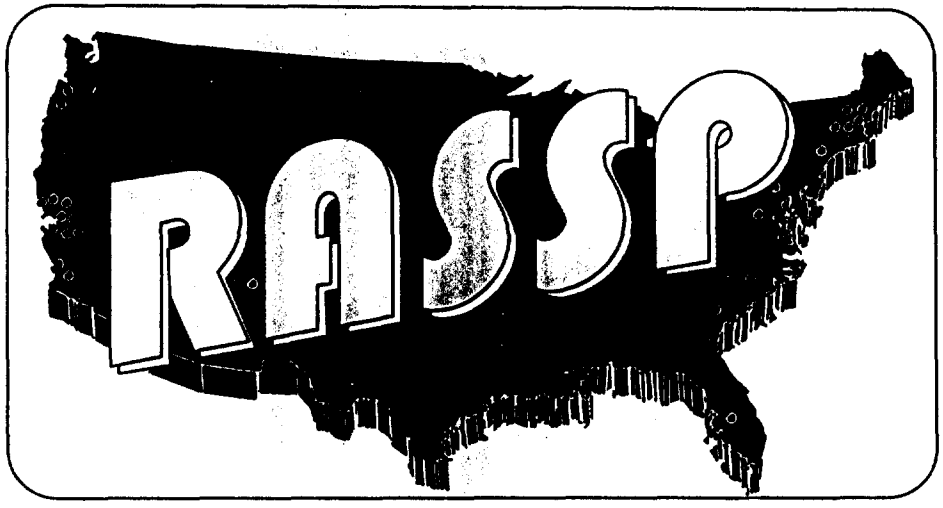
Contents

NRL/93-2172/93ACSD-02

AD A274324

Rapid Prototyping of Application Specific Signal Processors (RASSP)

Annual CAD System Description *FINAL*



23 December 1993

Submitted by: Lockheed Sanders, Inc.
Hughes Aircraft
Motorola
ISX Corporation

Under Contract: N00014-93-C-2172
As: CLIN 0004AG, Seq. No. A007

For: **Advanced Research Projects Agency**
3701 North Fairfax Drive
Arlington, VA 22203-1714

Naval Research Laboratory
4555 Overlook Avenue
Washington, DC 20357-5347

Lockheed Sanders, Inc.
P.O. Box 868
Nashua, NH 03061-0868

Approved for public release; distribution is unlimited.

List of Figures & Tables

Accesion For
NTIS CRA&I
DTIC TAB
Unannounced
Justification
By
Distribution /
Availability Codes
Dist | Avail and / or Special
A-1

Preface

This report is the final CAD System Description Report (A007), being submitted as a contract data item by the Lockheed/Hughes/Motorola/ISX Team under the RASSP Program.

Introduction

The goal of the RASSP Design Environment (RDE) is to automate the RASSP Product Development process, thereby contributing to its dramatic improvement in product development, specifically with respect to cycle time, product cost, and product quality. The RDE will provide technologies that fully support concurrent design, development, and electronic exchange of product information. The RDE is composed of three primary elements: a hardware component, a software component, and a data architecture. The hardware component consists of heterogeneous computers and high-speed data communication networks. The software component consists of computer-aided design (CAD) tools and an infrastructure within which the tools are integrated. The data architecture, known as the RASSP Engineering Database (REDB), is a distributed database containing the product life-cycle data for RASSP designs. End users of the RDE will see these elements as a single integrated computing environment supporting their product development activities across the entire product life cycle.

This document, the CAD System Description, is organized as follows:

- Chapter 1 presents an overview of the initial design environment, the mechanisms by which the design environment has evolved to this point, and the correlation between the design environment and the Rapid Development Group methodology.

- Chapter 2 presents specific information regarding each of the tools in the environment, including a description of the tools role in the environment, an assessment of strengths and weaknesses, and manufacturer's contact information.

The Lockheed Sanders Team, consisting of Lockheed Sanders, Motorola, Hughes Aircraft Company, and ISX, worked together in collecting the information for this CAD System Description document. Though documents exist that provide a similar description of tools used to develop Signal Processing systems, such as the Berkely Design Technology (BDT), the information in this document came from other sources. The majority of these sources include tool vendors and personnel from our own companies that have used these tools and are able to provide objective information on the strengths and weaknesses of these tools.

This document is planned to be available not only through hard copy, but through an electronic hypertext interface. In the first release of the RASSP Design Environment, this document will be accessible through the help viewer, permitting easy access to this information.

Section 1: The Initial RASSP Design Environment

The RASSP Initial Design Environment is identically the same as the environment currently in use by the Lockheed Sanders Rapid Development Group (RDG). The environment has been incrementally constructed over a period of five years as the RDG has striven to create a state-of-the art design environment suitable for the rapid prototyping and development of signal processing systems.

The Initial Design Environment consists of nearly forty tools covering most portions of the twenty-one categories of tools employed by the RDG. Since no single tool solves all problems in a given tool category, multiple tools are often used in a single phase of the development methodology to allow the best tool to be chosen for the specific problem being solved.The Initial Design Environment has not arisen from a static selection of current state-of-the art tools, but has rather evolved over the life of the Rapid Development Group. This evolution has been guided by the two following mechanisms which characterize the process for selecting tools to become a portion of this environment:

1) Areas of shortfall in the design environment's support for the development methodology have been identified. After identification of shortcomings, requirements for a better tool have been defined, and tools which may provide the needed functionality have been surveyed to determine how well they fill the need, if they are cost-effective, and if they can be readily integrated into the development environment.

2) New tools have appeared on the market which purport to solve problems in a better manner than existing tools. In an effort to proceed with a state-of-the-art toolset, these new tools, and their competitors, have been evaluated to determine if the tool is a good prospect for addition to the development environment.

All selected tools have been chosen because the development engineers using the design environment are of the opinion that the tools support the development process, and would aid in the automation of that process by increasing quality, productivity, or development cycle time.

The initial design environment is simply a "snapshot" of the current state of a continually evolving design environment. The environment must continually evolve to be successful, both to support an emerging development methodology, and to make the best use of state-of-the art tools and technologies. The Initial Design Environment tool set can be portrayed as a series of tools which support a development methodology. Figure 1.1 depicts the development methodology of the Rapid Development Group, progressing from Requirements and Specification through to Integration and Checkout of the system. The Initial Design Environment provides for a number of tool choices in some phases of the development methodology, while other phases of the methodology are still performed manually due to the lack of specific tools tailored to that phase of the development process.

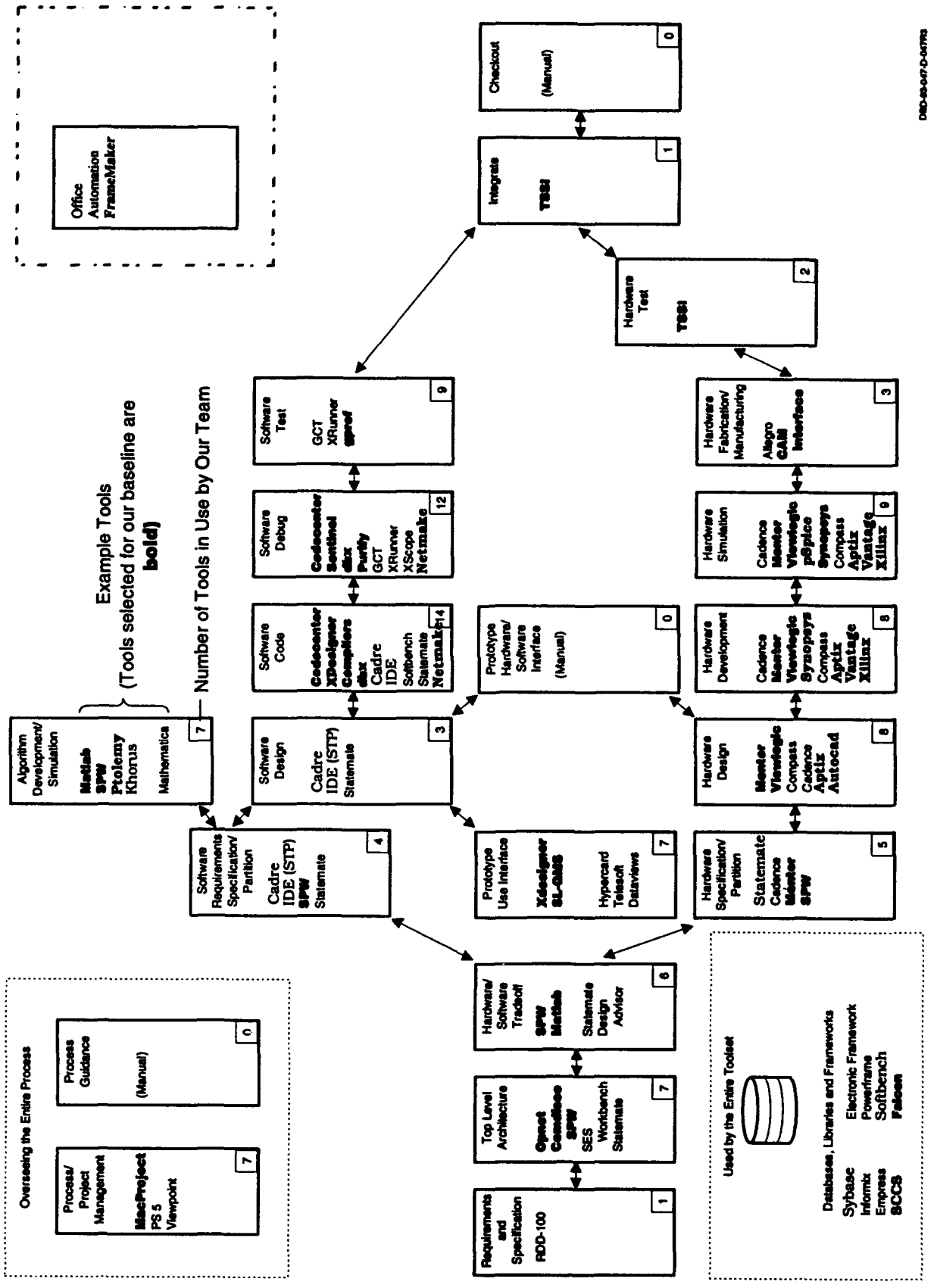


Figure 1-1: Initial Design Environment Tool Set and Process

INTEROPERABILITY OF TOOLS IN RASSP

The tools described in this report comprise our Initial Design Environment. The Lockheed Sanders Team will provide interoperability between these tools and additional tools in future releases of the RASSP Design Environment (RDE). This interoperability will be provided by utilizing numerous techniques for achieving interoperability, including common graphical user interfaces, common databases and translators, and evolving industry standards such as CFI.

Interoperability Through The Graphical User Interface:

The first high level form of interoperability is through the X-Window system. Signal processing tools are typically controlled by graphical user interfaces. A majority of these tools are X-Window compatible, but some run only on PCs or MacIntoshes. We have chosen to have the backbone of our computer network to be UNIX™ workstation based. Each engineer sits at a monitor that is capable of X-Windows, whether it be the console to a computer, or simply an X-terminal that boots off of a remote computer on the network.

This allows us to conveniently run most programs on the UNIX™ platform as they were intended, but also gives us the ability to run PC programs through emulators on the UNIX™ workstations.

In release of the RDE 0.1, there is a common graphical user interface that presents the user with the tools that are available to them from a menu. The user can select to run the tool by pressing the appropriate icon. This will make it much easier for the user to select a tool since he will not have to remember the exact syntax of the program name, start-up options, tool directory, or the computer to run from to run the tool. In future releases of the RDE, there will also exist a work flow program that will allow the user to choose the next function in design process to launch the tool.

Interoperability Through Database and Translators:

Our design environment stores project information in a database that is common between all tools. This database will not take the place of a specific tools database, but will keep track of the where the files are in the tools database and their versions, revisions, and other pertinent information. For example, a program may create a design file from a schematic capture tool. The information about the file will be stored in the database and other tools may access this file by translators that are automatically invoked in a manner which is transparent to the user. The information from the data stored will be used to derive data that can be used by dissimilar tools. For example, design information may be used to derive information needed by a costing tool.

Being able to access a database and translate the data into another tool not only helps several groups or companies to work together in the environment of their choice, but enhances reuse by making it easier for a project to access previous work done.

The RDE database is defined by the interface of required functions, so that any database may be used as an implementation. We are implementing the RDE with both an object database and a relational database so that we do not fix our design on one method to give us maximum flexibility.

The database will reside on a UNIX™ workstation using a client server architecture. The clients may exist in local UNIX™ workstations or remote workstations that connect from across the country. The clients may also be PCs or MacIntosh computers.

Future releases of the RDE will allow the data stored in the RDE database to trigger an event that causes another tool to be initiated automatically. This will be useful to run a program that notifies those involved with the next phase of the process of the results or status as a result of the changing data in the database. For example, reviewers of a design may be notified when the developer completes a design. Or a limit checking tool may be run automatically and notify the lead engineer when a design has changed sufficiently to cause more power, weight, or cost on a board design than is acceptable.

Interoperability through standards such as CFI:

To facilitate the integration of tools into the RASSP environment, we will make use of standards such as the CAD Framework Initiative (CFI) whose mission is to provide industry-accepted standards and technology that enable interoperability of tools and data for CAD systems. CFI members include many of the major computer and CAD system vendors. The use of the standards such as those provided by CFI will assist in the success of integrating tools into the RDE.

Section 2: Tool Descriptions

This document provides, for each of the computer-aided design (CAD) tools in the initial baseline RASSP design environment, a description of its role in the environment, an assessment of strengths and weaknesses, and manufacturer's contact information. This RASSP initial design environment is summarized in Table 1 which specifies the tool name, the manufacturer, and a short description of each tool. The detailed description and assessment of each tool comprises the remainder of this document following the introduction.

All of the tools listed have been assessed to have common functionality as opposed to unique functionality. An example of a tool with unique functionality is a signal processor synthesis tool. An example of a tool with common functionality is a schematic capture tool. All the tools listed have common functionality since they are not specific to a particular application. For example, Matlab is a tool that allows numerical computations of any mathematical description whether the application be electrical or mechanical.

Table 1: Initial Design Environment

Tool Name	Manufacturer	Description
Netmake	Aggregate Comp.	Software Development
Aptix	Aptix	Reconfigurable wiring
Autocad	Autodesk	3D Mechanical engineering
AutoTherm	Mentor	Thermal mechanical analysis
CodeCenter	Centerline	C code development
dbx/dbxtool	UNIX™	C debuggers
Falcon	Mentor	ECAD Framework
FrameMaker	Frame Tech	Word Processing
gcc	GNU	C/C++ compiler
gprof	GNU	Software profiling tool
Macproject	Claris	Project Management
Make	UNIX™	Software Development
Matlab	Mathworks	Algorithm Development
Mentor ECAD Sys	Mentor	Schmtc capture/layout/sim
ObjectCenter	CenterLine	C++ code development
Opnet	MIL3, Inc.	Network Performance Modeling
pSpice	MicroSim	Analog simulation
Purify	Pure Software	Software leak detection
Quad Design	Mentor	Signal integrity analysis
Sentinel	Sentinel	Software leak detection
SL-GMS	SL Corp	OO Graphical modeling system

Table 1: Initial Design Environment

Tool Name	Manufacturer	Description
SCCS	UNIX™	Source Code Control System
SPW/CGS	Comdisco	Algorithm Development
Sun cc	Sun	C compiler
Sun CC	Sun	C++ compiler
Synopsys	Synopsys	VHDL development environment
TDS Wavemaker	TSSI	Waveform/test vector generation
TDS Software	TSSI	Sys Design verify/analyze/test
Ptolemy	UC Berkeley	Algorithm Development
VHDL Simulator	Vantage	VHDL Simulator
XDesigner	VI Corp	X GUI Builder
PowerView	Viewlogic	Schematic capture/layout/simulator
VxWorks	Wind River	Real-time Operating System
Mathematica	Wolfram	Algorithm Development
Xact/Xilinx	Xilinx	FPGA Designer

Aptix (Aptix)

Synopsis:

Field Programmable Circuit Board Development System

Category:

Hardware Development Tools

Description:

The Aptix Field Programmable Circuit Board Development System provides a fully automated CAE environment in which hardware prototypes may be quickly configured, debugged, and changed.

Strengths:

- This tool allows prototypes to be built from schematics and debugged quickly. End result is faster and more thorough verification, as well as earlier product integration, and accelerated time to market.
- The tool may be integrated with a third party schematic entry tool as well as with a logic analyzer.
- Changes made to the schematics may be quickly implemented at the board level.
- Consistent documentation is ensured since the board database is always in agreement with the schematics.
- Routing may be performed incrementally to accommodate simple changes without affecting overall timing.
- Critical nets and pins may be specified for priority in routing.
- Step by step debug is possible since components may be electrically isolated, enabling small portions of the board to be wired at a time.
- Full signal observability is realized, since any of the signals may be brought out to be displayed on the logic analyzer.
- If used with programmable memory, logic, and microprocessors, it is possible to build hardware systems that are entirely programmable.

Weaknesses:

- The automatic partitioning and placement tool does not work well. The parts do not get partioned correctly and a lot of the parts do not get placed. The alternative is to use the manual placement tool, which works without problems and is generally the preferred method.

- There were some occasional problems when trying to route signals to the diagnostic port. If the group of signals to be observed reaches some critical number, the tool is not always able to route all of the signals. This problem could be resolved by deleting some of the signals in the list.

- There were some problems involving the ownership and protections placed on most of the files created while using the tool, creating problems if someone other than the person who created the design wished to work on the design.

- Some file editing may have to be done in order to get the schematic database in an Aptix - compatible format.

- There are limitations on the size and the speed of the designs that can be implemented using the Aptix system.

Overview:

The tool is relatively new, so there are some bugs that need to be worked out, though none that inhibit the immediate usefulness of the tool. Overall, it has proven to be a very useful tool with which to build and test prototype boards.

Version: 1.1a

Vendor Contact Information:

Aptix Corporation
225 Charcot Ave.
San Jose, California 95131
Phone: (408) 428 - 6200
Fax: (408) 944 - 0646

Installation Path:

/local/tools/aptix-1.1a

Invocation:

A script called 'aptix-1.1a' is used to invoke the tool.

The tool runs under openwindows. A user called aptix was created and the tool was run using this user name to avoid the file protection problem.

Licensing:

Currently one license node locked to sneezy

Vendor Documentation:

The documentation consists of a small user's manual. The manual is well written, easy to understand, and gives a good overview of the entire development process. It covers everything from hooking up the equipment to going thru a demo step by step. The demo comes already mounted on the board.

Items of Interest:

Thus far, the company has been very supportive in helping solve any problems encountered with their development system.

Autocad (Autodesk)

Synopsis:

Autocad is a computer-aided design (CAD) tool for 2D and 3D design and drafting.

Category:

3D Mechanical Design

Description:

Autocad is a mechanical design tool that one can use to create professional quality 2D and 3D models.

Strengths:

- Graphical User Interface
- Low cost
- Drafting capabilities
- 3 D capabilities
- 32 bit display space
- WYSIWYG plot preview
- Drawing Exchange Format (DXF) standard and IGES files for interchanging drawings between tools
- Work on multiple drawings simultaneously
- Compatible file format across platforms
- Over 200 add-on programs available for manufacturing functions

Weaknesses:

- Poor 3D surface modeling

Overview:

Autocad is a full-fledged drafting package for 2D and 3D drawings. The functions can be customized with the AutoLISP programming language to meet the user's design and manufacturing processes. The drawing files are compatible across platforms and other CAD software. A major weakness of the product is the 3D modeling capabilities not being as advanced as other CAD tools.

Version: 12

Vendor Contact Information:

Autodesk, Inc.
2320 Marinship Way

Sausalito, CA 94965
(415) 332-2344
(800) 445-5415
fax: (415) 331-8093

Installation Path:

Invocation:

Licensing:

Vendor Documentation:

Items of Interest:

AutoTherm (Mentor)

Synopsis:

CAD Thermal Analysis Tool

Category:

Hardware Design

Description:

AutoTherm is a thermal analysis tool that uses finite element techniques to analyze the thermal characteristics of printed circuit boards. AutoTherm performs conduction, convection, and radiation analysis, as well as potential fluid flow analysis to help determine accurate convection heat transfer solutions. AutoTherm then displays the results of thermal and flow solutions as maps, graphs, or tabular reports. AutoTherm calculates heat transfer coefficients automatically from information you provide with each model.

Strengths:

The graphic display is impressive, but, it is intended more for a mechanical engineer, than an electrical engineer.

Weaknesses:

The analysis results are only as good as the operator inputs. The results are based on the worst case or estimated power consumption which in most cases, is estimated by the operator. The power consumption is not based on the electrical simulations.

Version:

Vendor Contact Information:

Gateway Marketing Center
Mentor Graphics Corporation
P.O. Box 5050
Wilsonville, Oregon 97070-5050
Voice: 800-547-3000
503-685-8000
Fax: 503-685-8001

Installation path:

Invocation:

Licensing:

C/C++ Compilers

Synopsis:

The set of C and C++ compilers

Category:

Software Code

Description:

There are currently several C and C++ compilers installed on the system:

gcc:
A free compiler from GNU. Will compile both C and C++ code.

Strengths:
- Free software.
- Will compile both C and C++ code.
- C compiler is generally rated at or above the level of Sun's C compiler for SPARC processors. Optimizations for Super Sparc processors not yet available.
- Will compile ANSI code for nearly all UNIX™ platforms.
- Exceptional public domain support.

Weaknesses:
- C++ compiler is a true compiler and does not use a Cfront as most C++ compilers do. This can result in tighter compilation, but also makes object files generated by gcc incompatible with ObjectCenter.

Sun cc:
Sun Microsystems' C compiler.

Strengths:
- Currently free with SunOS up to Solaris 1.1. On Solrais 2.0 and later, it is unbundled and is licensed on a per-user basis.

Weaknesses:
- Non-ANSI compiler.

- Sun CC:
  Sun Microsystems' C++ compiler.

Strengths:
v2.1:
- Compilation times are very fast compared to the other C++ compilers installed on the system. The licensing allows for unlimited users.

v3.1:
- Provides template support.

Weaknesses:
- Neither compiler currently works with NetMake.

v2.1:
- No template support.

v3.1:
- Compilation times are much slower.

CenterLine C:
CenterLine Software's C compiler

Strengths:
- Included with CodeCenter, CenterLine's C debugging environment.

Weaknesses:
- Product is too new for a proper evaluation.

Centerline CC:
- Centerline Software's C++ compiler

Strengths:
- Included with ObjectCenter, CenterLine's C++ debugging environment.
- Provides for improved debugging of object files within ObjectCenter.

Weaknesses:
- Does not work properly with.KEEP_STATE option of make.
- Compile times are much longer than other C++ compilers.
- Does not work with NetMake.

Overview:

The C compiler of choice is gcc. gcc is an ansi standard compiler which produce smaller executables than other C compilers. All the C compilers work with NetMake, a distributed compilation utility, and CodeCenter, the debugger of choice.

Selecting a C++ compiler is a more complex issue. Of the aforementioned compilers, there is not one which offers all of the features we desire:

- Object-level compatibility with ObjectCenter (our C++ debugger)
- Fast compilation times.
- Compatibility with the AT&T 3.1 standard.
- 4) NetMake compatibility.

CenterLine CC is currently the worst match, but they claim to have plans to fix each of these (hopefully) by their next release. Currently we are using Sun CC v2.1 because of the compilation speed and ObjectCenter compatibility.

Version:

gcc:	2.2.2, 2.3.1, 2.3.3, 2.4.5, 2.5.0
Sun cc:	1.0
Sun CC:	2.1, 3.1
Centerline CC:	

Vendor Contact information:

Sun CC/cc:
Sun Microsystems
2550 Garcia Avenue
Mountain View, CA 94043
(415) 960-1300
FAX: (415) 969-9131

gcc:
Free Software Foundation (GNU)
675 Mass Ave
Cambridge MA, 02139
(available by anonymous ftp from prep.ai.mit.edu)

CenterLine C/CC:
CenterLine Software, Inc.
Pam Watkins
10 Fawcett Street
Cambridge MA 02138-1110
Voice: (617) 498-3267
Fax: (617) 868-6655
Workgroup ID: 30169

Installation path:

gcc:	/usr/local/lib/gcc-lib/sparc-sun-sunos4.1/2.4.5
Sun cc:	/usr/lang/cc
Sun CC:	/usr/lang/CC-2.1, /usr/lang/CC
CenterLine CC:	/local/tools/CenterLine/bin/CC

Invocation:

Licensing:

gcc:	Unlimited users

Sun cc: Unlimited users
Sun CC: 2.1: Unlimited users, 3.1 Floating license, currently limited to 1 user
CenterLine CC: Floating license, currently limited to 1 user.
CenterLine C: Floating license, currently limited to 10 users.

Vendor documentation:

Items of interest:

A supported version of gcc is available from Cygnus Software, however support from the Internet community is generally better and also free.

Cadre Teamwork

Synopsis:

Software tool for structure design and analysis

Category:

Software Development tool

Description:

Cadre Teamwork CASE tools automate standard structured methods (such as SASD, real-time system analysis, and information modeling) using interactive computer graphics. Within this integrated environment, a user can monitor the entire project's progress, rigorously check for errors, and produce design documentation.

Strengths:

Some of the advantages offered by Cadre Teamwork include the following:

- Stand alone analysis and system design
- Shared common database and revision control system
- Graphical User environment
- Facilities for interfacing to other tools
- Data consistency checks
- Heterogeneous networks support
- Require less UNIX™ knowledge
- Easy to learn and use
- Compatible data formats
- Competitive pricing

Weaknesses:

- Poor Object-Oriented Design support
- Drawing capabilities can be improved (fonts, font sizes, shading options, border options, etc.)

Overview:

Teamwork is a self contained analysis and design system; this integrated package gives an overall structure to the analysis/design/modeling process.

Version:

Release 4.0

Vendor Contact Information:

Cadre Technologies Inc.
222 Richmond Street
Providence, Rhode Island 02903-9990

Installation path:

Invocation:

Licensing:

Teamwork supports node-based (tied to one CPU) and service-based (limited to concurrent usage) licenses. However, the license server is node-locked.

Vendor documentation:

Items of interest:

CodeCenter (Centerline)

Synopsis:

C debugging environment and C compiler

Category:

Software Debug

Description:

CodeCenter is an interactive C debugging/development environment. CodeCenter is effectively a C interpreter which allows the loading of uncompiled source code (or a mixture of source and object code) for debugging. This feature provides greatly enhanced run-time error checking.

Strengths:

CodeCenter is an extremely good source-level debugger. Breakpoints can be set anywhere in the program, and data can be viewed and modified at any time. Since it is an interpretive environment, expressions can be evaluated on the command line as if they were part of the running program.

A combination of source/object code can be loaded at once to improve run-time performance. Only suspect modules need be loaded in as source code due to the performance degradation which results from interpreting source code (see weaknesses).

CodeCenter reports run-time errors which would otherwise be difficult to locate. When an error is encountered, CodeCenter brings you to the source file and line where the error/warning occurred. Some errors CodeCenter can detect at run-time:

- Undefined/Questionable arithmetic operators
- Undefined/Illegal pointer operations
- Information lost during conversion/assignment
- Memory allocation warnings
- Bad argument warnings
- Using unset memory
- Addressing errors, such as array bounds violations

A powerful data browser is one of CodeCenter's most useful attributes. Data structures are graphically displayed along with lines representing data pointers. This allows the user to visually examine the state of a linked list in a representation similar to the way it would be drawn on paper.

CodeCenter includes a C compiler, but will work with object files generated with gcc and Sun cc.

Weaknesses:

The initial loading of source files can be very time-consuming. Once the source files are loaded, run-time performance can be very poor due to the interpretive nature of CodeCenter.

Overview:

Overall, CodeCenter is an extremely powerful C development environment. CodeCenter specializes in detecting run-time errors which are typically difficult to locate, and can dramatically cut down on the time it takes to locate and fix software bugs.

Version: 4.0.2

Vendor Contact information:

Pam Watkins
CenterLine Software, Inc.
10 Fawcett Street
Cambridge MA 02138-1110
Voice: (617) 498-3267
Fax: (617) 868-6655
Workgroup ID: 30169

Installation path:

/local/tools/CenterLine

Invocation:

/local/scripts/codecenter
Link to shell file /local/scripts/centerline.csh, which executes all CenterLine tools.

Licensing:

Floating license, currently limited to 10 users.

Vendor documentation:

Items of interest:

dbx/dbxtool (UNIX™)

Synopsis:

Simple source level debugger

Category:

Software Debug

Description:

Dbx is a source level debugging tool which can debug C, C++, FORTRAN and Pascal programs. It has a simple, text-based interface which allows the user to set breakpoints, single-step and examine data during program execution. Dbxtool is a graphical front-end to dbx which runs under X/Openwin.

Strengths:

- Allows source-level debugging
- Provides facilities for stepping through program execution while examining data
- Is capable of showing the user what line an error occurred on during execution outside dbx by examining a core file.
- Loads in executable very quickly

Weaknesses:

- Text interface is difficult to use and requires a lot of typing. Graphical interface is better, but still lacks many features of modern debuggers.
- Data displayed is often lumped together in an unreadable format
- Does not support non-fatal run-time error checking like the Centerline debuggers do

Overview:

Dbx/dbxtool are a simple set of source-level debuggers with limited features. The only advantage that dbx has over the newer, interpretive debuggers, is the speed at which it is capable of loading in the executable and starting a debugging session. This set of debuggers does not allow for advanced run-time error checking for memory access errors and non-fatal violations. They can be useful, however, for quickly debugging simple errors.

Version: 4.1.3.

Vendor Contact Information:

Sun Microsystems
2550 Garcia Avenue
Mountain View, CA 94043
(415) 960-1300
FAX: (415) 969-9131

Installation Path:

/usr/lang/dbx

Invocation:

dbx

Licensing:

Included with SunOS

Vendor Documentation:

Documented in man pages as well as SunOS reference manuals

Items of Interest:

Falcon (Mentor)

Synopsis:

Tool used for the integration and concurrent use of multi-vendor and proprietary design tools.

Category:

Hardware Design.

Description:

Tool integration and encapsulation framework.

Strengths:

- Common User Interface (CUI) provides a consistent user interface to all applications. The CUI supports OSF/Motif standards, but also runs in Sun's OpenWindows environment.
- The Design Manager simplifies data coordination, data management, and application invocation.
- The Decision System allows information sharing throughout the entire product development process.
- Common data-modeling techniques, using object-oriented data models, unite all applications.
- The Design Data Management System provides the low-level management of data objects, such as maintaining multiple versions and attributes for each object. The multi-window feature allows you to have several applications displayed on the work station at once.
- The BOLD on-line information and help system makes all Mentor Graphics documents available on-line on a single CD ROM. In addition to Mentor documentation an organization can put their own documentation on-line using the BOLD Composer for Framemaker.
- The AMPLE programming language is a C like language used for the integration and encapsulation of third party vendor software.
- 79 third party vendors already integrated into Falcon Framework using the Mentor OpenDoor policy.

- Mentor Graphics is a founding member of the CFI standard and is currently bringing Falcon Framework to the CFI standard.

Weaknesses:

- Of the 79 third party vendors only two are for Software specific development.

Overview:

Falcon Framework is an industry leader and standard for Hardware Tool integration and encapsulation. A common user interface provides the user a shorter learning curve and compliance with the CFI standard will allow tools to talk and share a common database.

Version:

Software Version 8.2

Vendor Contact Information:

Gateway Marketing Center
Mentor Graphics Corporation
P.O. Box 5050
Wilsonville, Oregon 97070-5050
Voice: 800-547-3000
503-685-8000
Fax: 503-685-8001

Installation path:

Invocation:

Licensing:

Both Floating and Node Locked licenses are available.

Vendor documentation:

Items of interest:

FrameMaker (Frame Tech)

Synopsis:

FrameMaker is an advanced publishing tool that integrates word processing, graphics, page layout, and book building.

Category:

Publishing software

Description:

FrameMaker is a publishing tool that one can use to design, write, illustrate, and produce professional quality documents and books in a freeform approach or in a structured manner.

Strengths:

- GUI displaying formatted text
- Word processing capabilities
- Format templates for structured documentation
- Supports *internal and external cross-references*
- Graphics support
- FrameMaker typeset mathematical equations
- Keyboard macros that memorizes a sequence of keystrokes
- Compatible file across all supporting platforms
- Better color support in FrameMaker 4

Weaknesses:

- Memory storage space is generally greater than other word processors.
- Unable to open large files (e.g., files larger than 50MB)
- More expensive than most word processors

Overview:

FrameMaker is a complete publishing tool that combines several different applications offering many useful features. These powerful and convenient features can easily justify disadvantages such as price and memory space.

Version: 3.1

Vendor Contact Information:

Frame Technology Corporation
1010 Rincon Circle

San Jose, CA 95131-9847

Frame Product Sales Information: 1800-U-4-FRAME
Frame Order Administration: (408) 433-3311, ext. 6161

Installation path:

Invocation:

Licensing:

Both floating and fixed licenses are available

Vendor documentation:

Items of interest:

gprof (GNU)

Synopsis:

Executable performance profiling utility

Category:

Software Debug

Description:

Gprof is a C/C++ profiling utility. When a program has been compiled with gprof, execution of the program produces an execution profile of the program. For each function, gprof displays execution times, the number of times it was called, which function(s) called it, and the function(s) it calls. From this data, it is easy to see which functions are getting called often, and how much of the total execution time they are using.

Strengths:

- Free software from Gnu
- Works with C++ and has an option to demangle function names
- Displays percentage of total execution time each function consumed
- Easy to configure and use

Weaknesses:

- Does not report on program coverage
- Does not have a graphical user interface like other profilers
- It is necessary to rebuild the entire program from scratch in order to profile it. When the profiling is done, the program must be rebuilt again to remove it.

Overview:

Gprof is an extremely useful tool for determining bottlenecks in software performance. By laying out call frequency charts, gprof can point directly to the functions which are called most often, allowing the user to reduce the number of calls or optimize these critical functions to improve performance.

Version:

Vendor Contact Information:

Free Software Foundation (GNU)

675 Mass Ave
Cambridge MA, 02139
(available by anonymous ftp from prep.ai.mit.edu)

Installation Path:

/usr/lang/gprof

Invocation:

gprof

Licensing:

Free from Gnu

Vendor Documentation:

Macproject (Claris)

Synopsis:

Software tool used for effective project planning and management

Category:

Project management tool

Description:

MacProject Pro is a project management package that assists in planning, controlling and communicating projects efficiently and effectively by providing functions for brainstorming a project plan, building a schedule, assigning resources and costs, and monitoring project status.

Strengths:

- Graphical representation of a project life cycle
- Supports work breakdown structure
- Organizes tasks and milestones into a schedule
- Automatic scheduling provides calculations for resource duration, slack times, the critical path, and project dates.
- Provides various tables for budget information, tasks dependencies, and resource maintenance

Weaknesses:

- It's difficult to tailor the drawings in the schedule chart for presentation purposes
- In a project with multiple sub-projects, MacProject does not provide the option for the user to group the tasks by sub-projects

Overview:

Overall, MacProject Pro is a good tool for managing and maintaining project data. Since there is no standardized method for project management, it is essential for a tool to provide all necessary functions and capabilities. The user should determine whether or not these functions/capabilities can adequately handle the amount of project information effectively. In most instances, MacProject Pro is sufficient.

Version: 2.1v3

Vendor Contact Information:

Claris Corporation

5201 Patrick Henry Drive
Box 58168
Santa Clara, California 95052-8168

Installation path:

Invocation:

Licensing:

Vendor documentation:

Items of interest:

Make (UNIX™)

Synopsis:

Compilation generation and maintenance utility

Category:

Software Code

Description:

Make is a compilation utility which resolves and remembers compilation dependencies to aid in the construction/reconstruction of software executables.

Strengths:

- Included with SunOS
- Maintains all program dependencies in order to rebuild only the bare minimum necessary to construct a current executable. Make will always ensure that any and all source code changes have been incorporated into the generated executable.
- Can be used to build libraries and install software

Weaknesses:

N/A

Overview:

Make is a necessary software utility. It facilitates the compilation of programs by worrying about dependencies and which files have changed since the last compilation.

Version: 4.1.3

Vendor Contact Information:

Sun Microsystems
2550 Garcia Avenue
Mountain View, CA 94043
(415) 960-1300
FAX: (415) 969-9131

Installation Path:

/usr/bin./make

Invocation:

make

Licensing:

Included with SunOS

Vendor Documentation:

Documented in man pages as well as SunOS reference manuals

Items of Interest:

Mathematica (Wolfram)

Synopsis:

Mathematica is a general purpose software system and language intended for mathematical and other applications.

Category:

Algorithm development tool.

Description:

Mathematica is a numeric and symbolic calculator, a visualization system, and a programming language. Mathematica can also be used as a modeling and data analysis environment, a system for representing knowledge, and a software platform. It can also be used to create interactive documents that mix text, graphics, and sound. Mathematica can be used as a control language for external programs and conversely as an embedded system called from within other programs.

Strengths:

- Perform both definite (symbolic or numeric) and indefinite (symbolic) integration of complex functions.
- Solve systems of equations both symbolically and numerically.
- Perform matrix manipulations.
- Perform complex polynomial manipulations symbolically.
- Built in programming capability to supplement the library of functions.
- Very nice graphics in two and three dimensions.
- Multiple platform support.

Weaknesses:

- No symbolic system simulation. i.e. no drag and drop boxes containing user defined or Mathematica defined functions.
- No automatic C code generation.

Overview:

Mathematica is a very good tool used in solving complex math and engineering problems by combining numerical computation with graphical analysis.

Version:

Mathematica Version 2

Vendor Contact Information:

Wolfram Research, Inc.
100 Trade Center Drive
Champaign, Illinois 61820-7237, USA
Voice: 217-398-0700
Fax: 217-398-0747
Email: info@wri.com

Installation path:

Invocation:

Licensing:

Floating

Vendor documentation:

Items of interest:

Mentor DSP Station (Mentor)

Synopsis:

Mentor DSP Station encompasses a wide variety of DSP tools from software development to hardware development.

This description is a summary of the industry report titled "Design Tools and Methodologies for DSP Systems" put together by Berkeley Design Technology, Inc.

Category:

DSP Algorithm and Hardware Development.

Description:

- **DSP Station:** Design environment for digital signal processing, featuring floating and fixed point simulation, code generation, and hardware synthesis. Set of tools for graphically and/or textually specifying DSP systems.
- **DSP Station Design Entry:** DSP Architect, Filter Architect, and DFL Editor design entry tools.
- **DSP Station Simulation and Optimization:** DSP lab simulation, analysis, and optimization tools.
- **DSP Station Simulation Interfaces:** VHDL gen and DFL2LSIM behavioral simulation language interfaces
- **DSP Station Synthesis:** Code generation (C and Assembly) and hardware synthesis tools (Mistral 1 and 2).
- **System Design Station:** High-level requirement-driven system specification software.
- **GDT Designer:** Integrated circuit design tool for custom IC design.

Strengths:

DSP Station:

- Implementation independent system specification and design using DFL (Design Flow Language) provides the potential for hardware software co-design.
- Combination of filter design and optimization capabilities with MISTRAL 1 extremely useful for DSP IC design.
- Transaction log creates a record of all user actions allowing a rapid method of creating batch jobs.
- All basic library blocks provided as DFL source code which allows users to build new building blocks by adapting existing ones.

DSP Station Design Entry:

- Designs can be specified in DSP Station using a combination of three tools:

- DSP Architect - a version of Mentors Design Architect design capture tool enhanced to support DFL and DSP libraries.
- Filter Architect - a filter design system intended to design both II and FIR filters. Filter designs can be specified as frequency, phase, and impulse responses, as well as pole/zero plots and DFL code.
- DFL Editor - a text editor. Recognizes and helps the designer create syntactically correct DFL code. Support has been added for templates which allow users to write functions in C to be included within DFL description.
- All designs eventually reduced to DFL which facilitates both hardware and software design.

DSP Station Simulation and Optimization:

- Five tools integrated under DSPlab provides the user with the convenience of having all simulation tools under one environment.
- DSPview: a signal generation, display, and analysis tool. Allows the user to generate input signals and display and analyze output signals of a time domain simulation.
- TSIM: a time domain simulation tool. Generates and compiles C code from DFL and the resulting program is then used to simulate the system in the time domain. Simulations can be done using either fixed or floating point arithmetic. TSIM has the ability to switch between fixed and floating point simulation without reentering the design representation.
- FSIM: a frequency domain simulation tool.
- Cobra: a coefficient optimization tool. Cobra is used to minimize numeric precision to reduce implementation costs.
- Snake: a finite precision arithmetic analysis tool. Snake tests for limit cycles and coefficient sensitivities, and can also compute signal norms to aid in detecting overflow.

DSP Station Simulation Interfaces:

- Mentor provides an option for simulation by generating VHDL or M-language behavioral models for use by Mentors Lsim or QuickSim II. The DFL2LSIM and VHDLgen tools are used for this purpose.
- The CGEN package generates a C-description of the algorithm which can be compiled into assembly language code for most commercial DSP processors using the manufacturers.
- ASSYNT generates fully optimized and efficient assembly code for a specific signal processor. DSP Station options are ASSYNT_C30 for the Texas Instruments TMS320C30 and ASSYNT_M56 for the Motorola 56000.

DSP Station Synthesis:

- Contains CGEN (a C code generator), ASSYNT (an assembly code generator), and MISTRAL 1 and MISTRAL 2 (hardware synthesis packages).

- MISTRAL 2 is a powerful ASIC design tool allowing the user to experiment different architectures for implementing a particular algorithm or application.
- Generates optimized C and assembly code.

System Design Station:

- Performs system analysis based on structured analysis methodology.
- Develops state machines.
- Links graphical system analysis, prototyping and simulation environment together.
- Performs system simulation via VHDL and QuickSim II.

GDT Designer:

- Provides a set of tools for full-custom integrated circuit design including tools for symbolic layout capture, module generation, mixed-level simulation, and design verification.
- Provides the ability to realize very compact and fast chip layouts.
- Offers flexibility and support for design reuse.

Weaknesses:

DSP Station:

- Some elements are not well integrated with the Falcon Framework, and do not conform to the consistent user interface.
- Tools found to be sluggish even on a Sparcstation 2 with 80 Megabytes memory.
- The Berkeley group feels that as a whole, DSP Station has not reached the level of quality and reliability demanded by the serious user who will be paying between $40,000 and $200,000 for the system.
- Support for DSP Station is currently provided exclusively from Europe.

DSP Station Design Entry:

- DFL restricted to a single model of computation: synchronous data flow. This implies that DFL does not have the ability to easily describe control functions or asynchronous operations.

DSP Station Simulation and Optimization:

- Refer to generic DSP Station weaknesses.

DSP Station Simulation Interfaces:

- VHDL support seems to be weak and Mentor seems to be emphasizing Mentors "M-language".

DSP Station Synthesis:

- Poor microcode programming support for MISTRAL 2 to generate the same hardware for different DFL inputs.

System Design Station:

- Not fully developed yet.

GDT Designer:

- It would be desirable to integrate GDT with the DSP Station.
- Requires a large commitment of manpower.

Overview:

DSP Station and it's associated add-ons offer in one complete integrated environment, all the tools needed to design a system containing digital signal processing circuitry and its associated software.

Version:

Vendor Contact Information:

Headquarters:
Mentor Graphics Corp.
8005 S.W. Boeckman Rd.
Wilsonville, OR 97070-7777
Voice: (503) 685-7000
Fax: (503) 685-1202

Gateway Marketing Center
Mentor Graphics Corp.
P.O. Box 5050
Wilsonville, OR 97070-5050
Voice: (800) 547-3000
(503) 685-8000
Fax: (503) 685-8001

Installation path:

Invocation:

Licensing:

Standard Mentor Floating or Node Locked Licensing

Vendor documentation:

Items of interest:

Matlab (Mathworks)

Synopsis:

Graphical environment and programming language for numeric computation and visualization.

Category:

Algorithm development tool

Description:

MATLAB or MATrix LABoratory, is an interactive environment and complete programming language whose basic data element is a matrix of complex numbers that does not require dimensioning. This allows the user to solve numerical problems, implement algorithms and visualize results in a fraction of the time it would take to code the algorithm in FORTRAN or C.

Strengths:

Since the fundamental data type in MATLAB is a matrix of possibly complex entries, implementing complex numerical algorithms is extremely easy. MATLAB has a large library of built-in functions, all of which can operate on complex matrices. Instead of writing looped code, MATLAB's matrix operations encourage the user to implement algorithms in vector or matrix form, leading to compact code. Operations such as taking the Singular Value Decomposition of a matrix or taking the Fast Fourier Transform of a vector are reduced to a single line in MATLAB syntax.

MATLAB includes a large set of numeric functions as well as constructs for loops and conditionals. MATLAB also has an extensive set of built-in graphics functions that allow the user generate a wide variety of color plots and graphic effects. In addition, MATLAB also has functions for pull down menus, dialog boxes, slider bars and other objects so Graphical User Interface applications can be built in the MATLAB environment. A full suite of low level file input/output functions also exist to facilitate data exchange in arbitrary file formats.

One of MATLAB's most important features is its easy extensibility. New functions can be directly added to MATLAB by coding it in MATLAB syntax (M-files). In addition, MATLAB also provides the ability for C or FORTRAN code to be called directly from the MATLAB environment (MEX-files).

MATLAB offers a family of application-specific toolboxes or comprehensive collections of MATLAB functions that extend the MATLAB environment to solve specific classes of problems. Toolboxes exist in areas such as signal processing, control systems design,

approximation, dynamic systems simulation, system identification, neural networks and other areas.

MATLAB is ideal for algorithm development because it minimizes coding time and permits higher level thinking in algorithm development. It has become one of the standards in algorithm development for academia and industry. MATLAB runs on PC compatibles, Macintosh computers, as well as various workstations.

Weaknesses:

One of MATLAB's weaknesses occurs when it is run under the UNIX™ operating system. As variables are cleared and allocated by a MATLAB program, the memory is not returned to the UNIX™ system until the MATLAB process is terminated. This means that as a MATLAB script allocates and frees memory, the memory actually needed by the script
for variables can remain at a small value, but the amount taken up by the process can grow steadily until system problems occur. The only way to overcome this problem is to code memory-consuming MATLAB scripts carefully.

MATLAB code can be slower than C if the user implements algorithms as looped code instead of vectorizing it. Loops can be used, but their use should be minimized. In spite of this, MATLAB code still executes very rapidly. MATLAB's core routines such as eigenvalue/eigenvector solving, SVD, FFT, digital filtering, etc. are implemented directly in hand optimized C/assembly code on the host platform. Other functions that are not as computationally demanding are implemented using MATLAB syntax.

Other minor weaknesses of MATLAB are:

- MATLAB syntax cannot directly describe matrices with dimensions higher than 2. Third and higher order matrices must be implemented using single lower dimensional matrices.

- MATLAB can handle symbolic manipulation in the symbolic toolbox, but symbol manipulation is not as integrated into the environment as Mathematica.

- MATLAB also requires an annual license fee that is expensive, especially for a floating, multiple user license. The license fee must be paid every year or the license will expire and the tool will no longer run. A license fee is not required for PC and Macintosh platforms.

Overview:

A powerful and flexible programming and graphics environment for initial algorithm development and testing.

Version: 4.1

Vendor Contact Information:

Cathy DeYoung
The Math Works, Inc.
Cochituate Place
24 Prime Park Way
Natick, MA 01760

Voice: (508) 653-1415
Fax: (508) 653-2997
Email: tech@mathworks.comTechnical questions
suggest@mathworks.comProduct enhancement suggestions
bugs@mathworks.comBug reports
doc@mathworks.comDocumentation error reports
register@mathworks.comUser and product registration
service@mathworks.comOrder status, license renewals, passcodes
info@mathworks.comSales, pricing and general information

Installation Path:

/local/tools/matlab4.1

Invocation:

matlab41
- a local script that eventually calls /local/tools/matlab4.1/bin/matlab

Licensing:

Vendor Documentation:

Items of Interest:

Mentor ECAD System (Mentor)

Synopsis:

Mentor ECAD encompasses a wide variety of tools. This description focuses on the conventional suite of tools including schematic capture, simulation, and layout.

Category:

Hardware Development.

Description:

Design Architect: Is an integrated system for capturing electronic designs at the abstract architectural levels and the detailed logic and circuit levels.

IC Station: A family of tools for composite IC design whose flexibility adapts them to almost any IC design process.

Board Station: A comprehensive flexible system for printed circuit design and analysis. Unites the power and convenience of automated tools with the interactive expertise of the designer.

Idea Station: Design environment combining the Design Architect for multi-level creation and editing with QuickSim II for high-performance digital simulation and debugging. This integrated environment supports the entire top-down design and analysis cycle, from the abstract architectural level to the gate level.

QuickSim II: A mixed-mode, multilevel, high performance logic simulator capable of handling large designs with speed and precision for ASIC and PCB design.

MCM Station: A multichip module (MCM) design environment supporting the design process from design entry and analysis, through constraint-driven placement and routing, thermal and signal-integrity analyses, to manufacturing outputs and drawings. MCM Station supports all major substrate technologies, chip-mounting techniques, complex via structures, and data input and output formats commonly used in the design of MCMs.

Strengths:

Design Architect:

- Unified environment for top-down schematic and VHDL design creation.
- Integrated with Mentors Falcon Framework giving the user a consistent look and feel throughout all tools and also allowing the designer to use data from a variety of sources.

- The largest selection of digital, analog, and ASIC component libraries in the EDA industry.
- Allows creation of multiple configurations and annotations of all design elements from individual components to entire designs.
- Through parameter passing changes can propagate to every sublevel in which a characteristic occurs. For example you could change the delay on a hundred instances of a certain component in one step.
- Integrated VHDL editor maintains a variety of VHDL templates that help the novice learn the language and syntax.
- Optional VHDL compiler for synthesis and simulation allows immediate feedback on specific syntax errors in the VHDL editor.
- All product documentation is available on-line through the INFORM CD ROM technology. Search options and hyper-links help you navigate the entire manual set and instantly pinpoint the information you need.

IC Station:

- Integrated into Falcon Framework IC Station has tools for floor planning (ICplan), compaction (ICcompact), cell and block automatic place and route (ICblocks), design and electrical rules checking (ICrules), layout and schematic checking (ICtrace), layout parasitic extraction (ICextract), batch verification interface (Verification Dataport), and the User's own tools.
- Supports polygon, symbolic, cell, and block IC designs in combinations of unconstrained and connectivity based editing configurations.
- One database for all tools allows the designer to move freely from one tool or editing style to another without translating data.

Board Station:

- PCB layout, schematic capture, and simulation integrated into a complete electronic design automation environment with optional software for thermal and high speed analysis, and hybrid and multiwire technologies.
- Self learning tutorial course and a audio based personal learning program.
- Combines Computer Aided Engineering, Computer Aided Design, and Computer Aided Manufacturing in one Homogeneous Environment.
- Integrated into Mentors Falcon Framework providing an easy to use user interface based on the OSF/Motif standard.
- Ability to back-annotate key information to design files and schematic sheets for documentation, engineering changes, and simulation.

Idea Station:

- Integrated design creation and simulation environment reduces design time.
- Library support for major ASIC vendor and PCB simulation
- Design environment combines tools under a consistent user interface

- A component of the Mentor Graphics Concurrent Design Environment. The Falcon Framework for Concurrent Design extends the value of Idea Station by managing additional design, layout, and analysis data from other applications.

QuickSim II:

- Allows both behavioral and gate/switch level to be mixed in a simulation
- Advanced modeling using flexible, non-linear equations to calculate device and net delays improves performance and timing accuracy
- Integrates with VHDL simulation
- Fast design iteration with incremental design change eliminates the need to recompile/reload the entire design
- A component of the Mentor Graphics Concurrent Design Environment.

MCM Station:

- Constraint-driven layout allowing the user to specify timing, topology, electrical, and physical layout rules
- Powerful automatic and interactive high-speed place-and-route algorithms incorporates material property information such as layer thicknesses, dielectric constants, and conductivity
- Integrated thermal and signal integrity analysis tools
- Provides comprehensive post-layout analysis tools to check for wire lengths, special trace widths, clearances, connectivity, opens, and shorts against the schematic.

Weaknesses:

Overview:

Mentor ECAD system is a complete environment for the development of hardware from VLSI design to PCB layout. Mentor ECAD also provides various tools for simulation, synthesis, thermal analysis. This system integrates all tools under consistent user interface and database.

Version:

Vendor Contact Information:

Headquarters:
Mentor Graphics Corp.
8005 S.W. Boeckman Rd.
Wilsonville, OR 97070-7777
Phone: (503) 685-7000
Fax: (503) 685-1202

Gateway Marketing Center
Mentor Graphics Corp.

P.O. Box 5050
Wilsonville, OR 97070-5050
Phone: (800) 547-3000
(503) 685-8000
Fax: (503) 685-8001

Installation path:

Invocation:

Licensing:

Standard Mentor Floating or Node Locked Licensing

Vendor documentation:

Items of interest:

NetMake (Aggregate Comp.)

Synopsis:

Distributed compilation utility

Category:

Software Code

Description:

NetMake is a version of the UNIX™ make program that compiles separate modules of a large system in parallel on separate hosts. NetMake decreases compilation time dramatically by distributing processing across a number of networked workstations. For compilations where there are other workstations available on the network with the same architecture and operating system, significant performance enhancements are possible.

Strengths:

- Compilation times decrease markedly
- Easy to use

Weaknesses:

- A bug was found in that NetMake does not correctly handle targets with a fully-specified pathname (containing a / character) with the.KEEP_STATE option. However, the workaround has been found, and the NetMake customer support people devised a bug fix within 4 hours, to Federal Express us a new release the next day!
- Machines performing the "NetMake" must be of uniform architecture and operating system.
- Does not work with all compilers, most notably some C++ compilers.
- Will not schedule a job on a processor that is not relatively idle. This would ordinarily be an advantageous feature, but backfires when large jobs are "niced" and the CPU appears to not have much time to service NetMake jobs, but in actuality would take precedence over currently running jobs.

Overview:

NetMake is a powerful software tool which can greatly decrease compilation time, and therefore development time. The amount of performance increase is governed by the number of machines allotted to NetMake.

Version: 1.3

Vendor Contact Information:

Aggregate Computing, Inc.
300 South Highway 169
Suite 400
Minneapolis, MN 55426-1120
Phone: (800) 966-1666
Fax: (612) 546-9485
Email: support@aggregate.com

Installation Path:

/local/tools/netmake

Invocation:

Netmake

Licensing:

Floating license limited to 3 users

Vendor Documentation:

Items of Interest:

ObjectCenter (CenterLine)

Synopsis:

C++ debugging environment and C++ compiler

Category:

Software
Debug

Description:

ObjectCenter is an interactive C++ debugging/development environment. ObjectCenter is effectively a C++ interpreter which allows the loading of uncompiled source code (or a mixture of source and object code) for debugging. This feature provides greatly enhanced run-time error checking.

Strengths:

ObjectCenter is an extremely good source-level debugger. Breakpoints can be set anywhere in the program, and data can be viewed and modified at any time. Since it is an interpretive environment, expressions can be evaluated on the command line as if they were part of the running program.

A combination of source/object code can be loaded at once to improve run-time performance. Only suspect modules need be loaded in as source code due to the performance degradation which results from interpreting source code (see weaknesses).

ObjectCenter reports run-time errors which would otherwise be difficult to locate. When an error is encountered, ObjectCenter brings you to the source file and line where the error/warning occurred. Some errors ObjectCenter can detect at run-time:

- Undefined/Questionable arithmetic operators
- Undefined/Illegal pointer operations
- Information lost during conversion/assignment
- Memory allocation warnings
- Bad argument warnings
- Using unset memory
- Addressing errors, such as array bounds violations

ObjectCenter contains a useful C++ class browser which allows the user to examine the structure of loaded classes.

A powerful data browser is one of ObjectCenter's most useful attributes. Data structures are graphically displayed along with lines representing data pointers. This allows the user to visually examine the state of a linked list in a representation similar to the way it would be drawn on paper.

Weaknesses:

The initial loading of source files can be very time-consuming. Once the source files are loaded, run-time performance can be very poor due to the interpretive nature of Object-Center.

Included with ObjectCenter is a V3.1 C++ compiler. Early experimentation with this compiler revealed several disturbing problems:

- The CenterLine CC did not work properly with make's.KEEP_STATE option..KEEP_STATE is responsible for storing dependency information in between compilations, and without it our entire source directory would rebuild after even the smallest source modification. CenterLine reports that the next release (due out in October of 1993) will have resolved this problem.

- The CenterLine compiler's compile times were approximately four times slower than the C++ compiler we were currently using (Sun CC v2.1). CenterLine could offer no explanation as to the cause of this.

- CenterLine CC currently does not work with NetMake. CenterLine is currently engaged in negotiations with the manufacturers of NetMake to resolve this problem.

Overview:

Overall, ObjectCenter is an extremely powerful C++ development environment. Object-Center specializes in detecting run-time errors which are typically difficult to locate, and can dramatically cut down on the time it takes to locate and fix software bugs. The supplied compiler has some major drawbacks, but CenterLine seems to be committed to bringing it up to par.

Version: 2.0.2

Vendor Contact information:

Pam Watkins
CenterLine Software, Inc.
10 Fawcett Street
Cambridge MA 02138-1110
Voice: (617) 498-3267
Fax: (617) 868-6655
Workgroup ID: 30169

Installation path:

/local/tools/CenterLine

Invocation:

/local/scripts/objectcenter
Link to shell file /local/scripts/centerline.csh, which executes all CenterLine tools.

Licensing:

Floating license, currently limited to 1 user.

Vendor documentation:

Items of interest:

Opnet (MIL3, Inc.)

Synopsis:

Systems modeling, simulation and analysis tool.

Category:

Modeling and simulation tools.

Description:

OPNET is a systems level modeling, simulation and analysis environment. In particular, the OPNET environment is a collection of tools designed to model and simulate networks at a systems level. The paradigm of a network is made sufficiently general in OPNET so that it can accommodate a wide variety of problems of interest.

Strengths:

The OPNET environment supplies an integrated set of tools for the development of system models and their simulation. These tools form a hierarchical development environment within OPNET which ranges from the individual process level (lowest level) to the network level (highest level). The framework established by these tools allows the user to partition many problems in a natural way without forcing the user to follow strict rules regarding the order in which components of the model are developed.

Model development is supported by a limited set of templates which are supplied with the tool. The availability of these templates reduce model development time by allowing the user to copy and/or modify existing structures from other models for his or her own use. Examples include: queuing, routing, Ethernet, FDDI, TCP/IP, and token ring models.

Flexibility and fidelity in model development are aided by attribute and parameter editors which give the user the ability to independently bound system performance measures and define the resolution of the result. Flexibility is further augmented by OPNET's foundation in the C programming language, and its ability to be integrated with other C based systems.

Access to system parameters which are needed during simulations or for debugging purposes is provided via probe and debugging tools. The probe tool allows the user to tap data flows throughout the system for analysis at a later time. It also has some limited state transition animation capability at the process (lowest) level. Restricted interactive run time access to system parameters is available through the on line debugger (odb) supplied with the OPNET environment. Additional access to these parameters may be gained through interactive debuggers such as dbx (or xdbx) which may be attached to the compiled simulation.

Once simulations are compiled, they may be run outside of the OPNET environment on any computer which supports the executable code. This feature reduces the number of OPNET licenses required to only those needed for actual development. Separate licenses for executing the simulations are not required. Further, since system parameters can be collected into ASCII files which can be read by the simulations, simulations with different parameters, probe files, output files, etc. can be run simultaneously on different machines; greatly speeding up the process of bounding system performance.

Upon completion of a simulation, the analysis and filtering tools of the OPNET environment may be used to study the results. The analysis tool allows the user to view the numeric outputs of the simulation (e.g. bit throughput rates, packet size, utilization, etc.) in graphical form. It can organize these presentations in a variety of ways. Examples include histograms, cumulative density functions, and scatter plots. The user also has the ability to filter these results for customized displays. This is done using the filter tool. With the filter tool the user can construct a large number of filters of almost arbitrary type using OPNET supplied primitives. These primitives include adders, delays, integrators, differentiators, multipliers, etc. and can be bound together in various ways to produce the desired result (e.g. fir filters, iir filters, etc.).

Finally, the documentation for OPNET is extensive (11 volumes) and the technical support is excellent.

Weaknesses:

The templates supplied for model development are limited, and confined primarily to commercial communication protocols such as ethernet, TCP/IP, etc.. Widely used military protocols such as 1553 are not part of this package. This forces the user to develop much of the structural underpinnings of models of interest from scratch. This can be time consuming and the results may be difficult to benchmark.

The on line debugger (odb) supplied with the OPNET environment is limited in its utility relative to other debugging tools such as CodeCenter. Because of the proprietary nature of some of the OPNET systems software (e.g. the simulation kernal), and because of the length of time involved in some of the simulations, the user is often forced to resort to less comprehensive debuggers such as dbx (or xdbx) which can be attached to the executable simulation.

OPNET was originally designed to simulate networks. Although, in the OPNET paradigm, the concept of a network is rather general, not all systems may fit into their general notion of a network. This has not been a problem in the systems modeled to date, but it could be a limitation in some cases.

Overview:

The OPNET environment is a powerful tool for systems modeling, simulation and analysis. It is capable of detailed simulations, but is most effective when used as a high level demonstration/validation (DEMVAL) tool at the early stages of development for the purpose of risk reduction. If the models created are maintained and updated throughout the

project design cycle, then OPNET can be used effectively as a check on the (evolving) design.

Version:

2.4.A

Vendor Contact Information:

MIL 3, Inc.
3400 International Drive NW
Washington, DC 20008
Voice: (202) 364-8390
Fax: (202) 364-6182
Email: opnet@mil3.com

Installation path:

/auto/aries_2a/opnet

Invocation:

/local/scripts/opnet

Licensing:

Floating license, currently limited to one user.

Vendor documentation:

Eleven volume set.

Items of interest:

pSpice (MicroSim)

Synopsis:

The Design Center with schematic entry is a full-featured analog/digital design and simulation system.

Category:

Hardware Design

Description:

Integrated Analog and Digital Simulation

Strengths:

- Schematic capture with graphical circuit editing and a Symbol Library with over 6,700 parts.
- Hierarchical designs allowing you to create multiple representations for a given block of circuitry.
- Mixed analog and digital pSPICE circuit simulation.
- Graphical waveform analysis providing simultaneous viewing of analog waveforms and digital signals produced by the pSPICE simulations.
- Filter synthesis for realizing active and passive filters supporting RC active, switched-capacitor, and LC ladder architecture's.
- Signal integrity analysis for extracting transmission line, parasitic capacitance, and coupling values from printed circuit board layouts, which are then applied to simulations in pSPICE for analysis of crosstalk, reflection, and delay effects in circuit design.
- Stimulus generation for characterizing and creating analog and digital stimuli for the circuit.
- Device characterization for creating semiconductor device model and subcircuit definitions by estimating model parameters.
- Cross platform support from Microsoft Windows, DOS, HP9000/700, and Sun.
- pSPICE and the graphical waveform analyzer can be invoked by Mentors Design Architect under Falcon Framework.

Weaknesses:

- Entire Design Center not integrated into Mentor's Falcon Framework.
- Simulations not VHDL compatible.

Overview:

The Design Center EDA provides a fully integrated environment to capture, simulate, and analyze analog and digital circuitry.

Version: 5.4

Vendor Contact Information:

MicroSim Corporation
20 Fairbanks
Irvine, CA 92718
Phone: (714) 770-3022
(800) 245-3022
Fax: (714) 455-0544
BBS: (714) 454-7611

Installation path:

Invocation:

Licensing:

Vendor documentation:

Items of interest:

PowerView (Viewlogic)

Synopsis:

PowerView is a UNIX™-based electronic design environment which supports the design process through innovative tool technology, efficient project management, and a configurable modular architecture.

Category:

Hardware Development Tools

Description:

PowerView is an open design environment used for accessing all design tools, regardless of their purpose or manufacturer. The environment is centralized around a design cockpit, which is where PowerView executes and manages tools, controls libraries and projects, and reports messages. The PowerView framework encapsulates Viewlogic, third party, and user-developed design tools within a seamless environment, providing intertool communication and tool encapsulation to increase design efficiency and optimize design resources.

Strengths:

- A common user interface
- A method for incorporating tools into the framework
- A method for executing each tool
- A project manager
- Tools are well organized into groups related to the solution of a particular design task.
- Multiple tool windows allows the use of several tools at the same time.
- The environment may be custo ed and the menus may be modified as desired.
- Macros may be used to record the command sequence of repetitive tasks into a file.
- Most commands may be executed at the command line.
- Any software program can be assimilated into the powerview framework as a tool, including text editors, spreadsheet and database programs.

Weaknesses:

- The board database update feature does not work correctly when invoked from within an open Viewplace design. It should be invoked from the pop up menu which appears when the Viewplace tool is first opened.

- The Viewplace board database may become corrupted if a package is altered after the board database has been created.

Overview:

The latest version of the tool (PowerView as compared to Workview) has resulted in a completely overhauled and very centrally organized user interface. The tool does not require a great deal of time to learn the basics of operation, as many of the features are quite intuitive. PowerView is a very powerful tool when used for schematic entry and simulation. It also works fine for board placement, although the board placement tool is not one of its strongest features.

Version: 5.1.2

Vendor Contact Information:

Viewlogic Systems, Inc.
293 Boston Post Road West
Marlboro, MA. 01752
Phone: (508) 480 - 0881
Fax: (508) 480 - 0882

Installation Path:

/local/tools/powerview-5.1.2

Invocation:

A script called "powerview-5.1.2" is used to invoke the tool.

Licensing:

Currently there are two licenses.

Vendor Documentation:

All of the documentation is contained on line.

Items of Interest:

It may not be possible to open tool windows on designs from two different projects at the same time. It would be helpful if this was possible.

Ptolemy (UC Berkeley)

Synopsis:

Signal processing algorithm development environment

Category:

System Engineering

Description:

Ptolemy is an integrated algorithm development environment for algorithm design, algorithm test and simulation, hardware simulation, code generation, and parallel computing. Integration of these different aspects permits evolution from one design stage to the next.

Ptolemy is composed of several "domains", each with its own model of computation. A domain is composed of a scheduler and a number of functional blocks. The blocks are the basic unit of computation with the scheduler controlling their execution for a particular simulation. Various types of data are passed between blocks. Several domains can be mixed in one simulation. The domains are as follows:

- Synchronous Data Flow (SDF) - Data flow not dependent on data itself. Contains most of the predefined signal processing blocks. Simulation is scheduled at compile-time.

- Dynamic Data Flow (DDF) - This domain is a superset of SDF so all SDF blocks are available. Control is data-dependent, scheduling is finalized at run-time. Most simulations of "traditional"signal processing algorithms can be built with the SDF and DDF domains.

- Discrete Event (DE) - The DE domain is an environment for simulation of systems where the time-stamp of an event is important such as communication networks. Events represent changes in the state of the system and have a time associated with them.

- Thor - Thor is a mixed-level functional hardware simulator covering top-level behavior, register transfer, and gate levels. Thor is a timed domain, and information is time-stamped.

Code Generation (CG) - CG generates code as opposed to running a simulation. It includes a parallel scheduler for partitioning a generic task among some number of processors with a predefined communication cost across processors.

Strengths:

- The primary advantage of Ptolemy over most signal processing development environment is the ability to do heterogeneous system modeling. For example, a speech compression system that transfers its data over a packetized communications network can be modeled by a DDF simulation contained within a DE domain simulation. The speech coder can be implemented in the DDF domain, while the DE domain is suited to modeling the communications network. Inside, the speech coder has its own DDF scheduler, but outside the DDF domain (in the communication network simulation) it appears as a basic DE block and is treated that way by the overall DE scheduler. The key is the interface between domains which makes the DDF simulation appear to be a DE block.

- Ptolemy is an object-oriented software system that is programmed in C++. The core, or kernel, has a very general structure that imposes few constraints other than defining the interfaces for the different design methodologies. Data abstraction and information hiding are design concepts of the kernel that allow interactions between different models of computation. The design is intended to be extensible to new models.

- The graphical interface is reasonably easy to use. The block diagram is helpful for the system designer to explain the algorithm concept to others. It provides a clear visual of the flow of data.

- It is possible for the user to add new functional blocks. New blocks can be linked dynamically to the standard version of Ptolemy, thus avoiding recompilation; or the user can create his own version containing his blocks. A preprocessor assists by generating standard initialization code and formats and comments the code in a systematic manner. Given a knowledge of C++ and the block requirements for different domains, writing new blocks is reasonably straightforward.

Weaknesses:

- For this version, Thor is labeled as an "experimental" domain, as all the details of melding with the rest of Ptolemy have not been addressed.

- CG is experimental, and at this point, includes only a parallel scheduler for partitioning a generic task among some number of processors with a defined communication cost across processors. It does not actually produce code, except stand-alone C++ code can be grouped and compiled for SDF simulations only.

- The syntax of specifying an application in text is somewhat cumbersome. Also, the resulting list of commands can be difficult to read for someone other than the designer.

Overview:

Ptolemy is a design and simulation environment with capabilities that extend beyond basic signal processing simulation packages. It incorporates a graphical user interface to allow system specification with block diagrams. It has timed domains (DE) to accommodate simulations of systems such as communication networks, and a hardware simulation domain (Thor). Interfaces have been defined to combine multiple domains into a single simulation, a central objective of Ptolemy. A significant attribute is the code generation facilities, some of which are still under development. For "standard" signal processing simulations, simpler, typical packages would probably be quicker and more efficient for system design and specification. However, the mixed domain and code generation features of Ptolemy are an advancement when applied to appropriate problems

Version: 0.3.1

Vendor Contact Information:

Mary Stewart
Email: marys@diva.berkeley.edu

Edward Lee
Email: eal@ohm.berkeley.edu

Installation Path:

/local/tools/ptolemy

Invocation:

/local/tools/ptolemy/bin/pigi

Licensing:

No cost license from Berkeley, unlimited number of users.

Vendor documentation:

/local/tools/ptolemy/Almagest (troff)

Items of interest:

Purify (Pure Software)

Synopsis:

Software memory leak detection.

Category:

Software Debug

Description:

Purify is a run-time software memory leak detector which reports on possible memory leaks as the application runs, giving line numbers and call stack traces of the offending source. The Purify command is easily inserted into a makefile, and only a re-link is necessary to insert Purify into an application binary.

Strengths:

Easy to add to or remove from an application.

Provides excellent detection of the following problems:

- Memory leaks
- Array bounds errors
- Reading/writing freed memory
- Freeing memory multiple times
- Using uninitialized memory
- Accessing null pointers
- Stack overflow due to overly recursive function calls

Supports many C/C++ compilers

Weaknesses:

Purify is rather expensive. In fact it is ten times more expensive than another comparable leak detection package.

Creates many support files in the application directory while linking.

Overview:

Overall, Purify performs as advertised. Purify specializes in detecting errors which are generally difficult to locate by hand, and is an important software development tool. The main drawback is the relatively high cost of the software compared to other memory leak detection packages.

Version: 1.1.2 (06/26/92)

Vendor Contact information:

Pure Software
1309 South Mary Avenue
Sunnyvale, CA 94087
Phone: (408) 720-1600
Email: support@pure.com

Installation path:

/local/tools/purify

Invocation:

In Makefile, place /local/tools/purify/purify in front of linker on link line.

Licensing:

Floating license, currently limited to 1 user.

Vendor documentation:

Items of interest:

Quad Design (Mentor)

Synopsis:

Mentor Quad Design provides several products that can be used to determine layout effects on PCB and MCM designs.

Category:

Signal integrity analysis tool

Description:

XTK (Crosstalk Tool Kit) - network simulation/analysis tool that can be used to predict system level noise and interconnect effects prior to costly fabrication. XTK analyzes linear/non-linear effects, loss, and crosstalk contributors.

XNS (Crosstalk Network Simulator) - performs exhaustive simulation of the effects of crosstalk on transmission lines in digital systems. In addition to outputting a report listing the maximum net to net noise contributions, XNS produces a waveform display that can be manipulated by the user for detailed signal quality analysis.

XFX (Crosstalk Field Extractor) - calculates electrical parameters such as dynamic impedance, trace velocity, capacitance, inductance, resistance and coupling of arbitrary configurations of conductors, ground planes and dielectrics. XFX uses non-linear spatial warping and relaxation techniques to compute the effects of all fringing fields on the computed transmission line parameters.

PDQ (Pre-route Delay Quantifier) - PCB placement analysis tool estimating interconnect length, loading and delay prior to routing. PDQ identifies critical nets, interconnect delay problems, and placement-induced delay problems.

MOTIVE (Modular TIming VErifier) - identifies all set-up and hold violations in a design by exhaustively tracing every signal delay path. MOTIVE also details the component delay, interconnect delay and slack time for both the clock and data path that caused the violation.

TLC (Transmission Line Calculator) - simulates and displays the effects of transmission line phenomena on digital systems. TLC predicts ringing, undershoot and overshoot and offers delay data for all digital technologies.

XFX3D (Three-Dimensional Field Solver) - models three dimensional structures; simulates packaging/interconnect parasitics prior to manufacturing, analyzes the electrical consequences of PCB/MCM manufacturing rules on circuit parasitics, and validates alternative connection strategies.

Strengths:

- provides extensive analysis of signal integrity behavior during electronic design stage.
- compatible with a variety of CAD tools such as Gerber Photoplot, Cadence Allegro, Calay VO4 and Prisma, Cooper & Chyan Spectra, MentorGraphics Boardstation, SCICARDS, PADS PCB, PCAD PDIF, Racal-Redac, and VIEWlogic VIEWplace.
- various CAE interfaces include Calay, Cadnetix, CASE Vanguard, FutureNet, Mentor Graphics, PCAD PDIF, SCICARDS, TDL, Cadence, Valid, VIEWlogic, Verilog, EDIF, Racal-Redac, PADS.
- various platform support
- able to perform distributed processing of large jobs. The distribution can be performed across different platforms.
- ASCII format component libraries

Weaknesses:

- does not handle analog signal at all (such as a sinusoidal source)
- no analog design support
- feedback loop glitches
- does not use SPICE simulator; the user must acquire data from vendors to build a library of parts
- non-parallel lines are not included in the couplings even if the lines cross
- does not handle temperature dependencies
- integration to other layout/field solver tools can be improved

Overview:

Quad Design offers a powerful suite of system-level timing and signal integrity analysis tools. These tools allows the designers to exhaustively analyze circuit or system timing and signal integrity behavior without building prototype boards. Overall, Quad Design can reduce design iterations, thus improve quality and time to market.

Version:

XTK (XNS, XFX, and TLC) 5.02
MOTIVE 4.02
XFX3D 1.0.2
PDQ 2.0.7

Vendor Contact information:

1385 Del Norte Road
Camarillo, CA 93010
Voice (805) 988-8250
FAX (805) 988-8259

Installation path:

Invocation:

Licensing:

Floating licenses available

Vendor documentation:

Items of interest:

SCCS (UNIX™)

Synopsis:

Source code control system

Category:

Software Code

Description:

SCCS is the configuration management tool supplied with the Sun operating system. It manages software source code in a central repository and keeps track of versioning in a multiuser environment. SCCS allows several people to work with a set of software source files by preventing concurrent file modification. When a user wishes to edit a source file, he "checks out" the file using SCCS, makes his modifications, and then "checks in" the file. During the period of time the user has the file checked out, no other user may check the file out.

When a user checks a file back in to the repository, the version number of the file is incremented. SCCS supports the retrieval of previous versions which are stored in the repository in a delta format.

Strengths:

- Successfully prevents multiple people from modifying the same source file
- Only saves changes to a file rather than storing an entire new file
- Included with SunOS
- Supports versioning
- Supports diffs on previous versions

Weaknesses:

- Does not support parallel development model
- It is very easy to circumvent SCCS's CM procedures
- Does not contain any security features. Users are essentially trusted to follow the SCCS procedures.

Overview:

SCCS is very useful for simple source code control, but lacks support for parallel development. The SCCS paradigm is fundamentally incorrect in its assumption that during software development, the case will never arise when more than one user needs to modify the same source file. In practice, this often happens with header files which are shared between modules.

For simple source code control, SCCS works just fine, maintaining source versions in an optimal format, and allowing several user to access the same source code repository.

Version:

Vendor Contact Information:

Sun Microsystems
2550 Garcia Avenue
Mountain View, CA 94043
(415) 960-1300
FAX: (415) 969-9131

Installation Path:

/usr/ucb/sccs

Invocation:

sccs

Licensing:

Included with SunOS

Vendor Documentation:

Documented in man pages as well as SunOS reference manuals

Items of Interest:

SL-GMS (SL Corp)

Synopsis:

C model builder

Category:

Prototype User Interface

Description:

The SL-GMS system provides a very powerful set of capabilities for creating, updating and displaying dynamically graphical objects.

Strengths:

- Source is provided for many demo programs. Although hard to understand initially, these demos provide a large sample of working code to modify to suit specific needs.
- Each individual model can be included in another model with minimal effort. (A hierarchical design is possible.)
- It can be integrated with X-motif widgets, or other third-party library-based toolkits.
- A new SL-DRAW editor is in beta-test; nearly all of the below complaints about SL-DRAW have been fixed in the new version. (The beta-test version is included in our software, and seems to correct nearly all the SL-DRAW user-interface problems mentioned below.)
- The tool is initially easy to use; in drawing of shapes, figures, or text entry, a prompt continually identifies what type of input is expected next.

Weaknesses:

- The "percent fill" function only works for rectangles. Other closed objects (circles, sectors, ellipses, splines) cannot be filled to a given percentage.
- It is not clear how (if it is even possible) to set up external datasources to "trigger" an event in gms, outside of having gms poll the external datasource at a regular frequency.
- A graph with many more points displayed than pixels, updated at a high speed (multiple times per second) causes some flicker with the graph. However, such a graph is of questionable desirability for other reasons.

- Graphs are created using SL-GML, a modeling language, rather than using the SL-DRAW tool. This requires the programmer to learn and understand another tool in order to create or modify graphs.
- Many aspects of the SL-DRAW tool are counterintuitive.
- Reshaping of some figures, such as a sector or graphs, are not permitted.
- Currently selected objects are not clearly marked (manual indicates they should be).
- Order of display is supposed to be settable; this does not appear to have any effect.
- The correspondence between button presses and attributes assigned to an object is not clear; yet, seems to be important for dynamic simulations.
- Pull-downs, buttons, and textual labels are indistinguishable.
- In pull-downs, the current choice is not highlighted as the cursor passes over it.

Overview:

Overall, SL-GMS would be a powerful tool to the programmer, but will require a large start-up time for a programmer to understand how to use the drawing editor, dynamic language, and library functions.

Version: 4.1.3

Vendor contact information:

Installation path:

Invocation:

gms/bin/draw to draw the desired interface, then find an example in gms/demo that resembles the tool being constructed and modify to fit the new model.

Licensing:

Vendor documentation:

Manuals are poorly organized:

- indexes of manual are minimal in content, cross-references
- separate tools are not described separately
- an overview or "getting started" section is lacking
- examples introduce new ideas/concepts with no explanation

- dynamic variables of gismos are not explained in depth for all objects

Items of interests:

SPW/CGS (Comdisco)

Synopsis:

Software tool for DSP, communication, and hardware design

Category:

DSP Development tool

Description:

SPW is a graphical programming tool supporting digital logic and signal processing design. This integrated framework includes tools such as the Code Generation System (CGS), the Hardware Design System (HDS), the Designer/Block Diagram Editor (BDE), Signal Calculator, ProCoder, Filter Design System (FDS), and MultiProx (MPX). SPW can be used for algorithm development and simulation, filter design, hardware design, analysis, and synthesis.

CGS

Code Generation System generates C code from a SPW block diagram specification of a system. The code can be used for simulation acceleration, application building, or real-time prototyping.

Strengths:

SPW is a powerful DSP tool. Some features are:

- Graphical programming environment
- Signal processing software library
- C, assembly code and netlist generation from DSP block diagram
- SPW Design Database for resource management
- Multi-level macro design
- HDS VHDL simulation -- compatible with other simulators (e.g., Mentor Graphics and Synopsys)

CGS

- Simple to generate a C code version of the system from a floating-point SPW simulation.
- eliminates dead code thus reducing the code size

Weaknesses:

Some drawbacks of SPW include the following:

- SPW's software development environment does not support basic programming language constructs such as conditional statements (if-else), case statements, and loop structures. These features are essential for a software design environment.
- SPW does not have an equivalent ASIC development tool such as Mentor's Mistral2.
- During simulation, SPW updates all data at each clock pulse. This is less efficient because some data may not require the recomputation.
- SPW makes clear distinction between fixed-point and floating-point blocks. This characteristic often results in multiple designs achieving the same functionality.
- SPW does not have a simple function for the designer to examine simulation output at a node by specifying the node name.

CGS

- Cannot be used with HDS fixed-point blocks

Overview:

SPW is an excellent tool for DSP, communication, and hardware design. The emphasis of this tool is placed on the hardware environment more than the software. Tools provided by this DSP framework are integrated nicely under a consistent user interface and database.

Version: 3.0

Vendor Contact Information:

Comdisco Systems, Inc.
919 E. Hillsdale Blvd.
Foster City, CA 94404
Voice: (415) 574-5800
Fax: (415) 358-3601

Installation path:

Invocation:

Licensing:

Node-locked and floating licenses available.

Vendor documentation:

Items of interest:

Sentinel (Sentinel)

Synopsis:

Software memory leak detection.

Category:

Software Debug

Description:

Sentinel is a run-time software memory leak detector which reports on possible memory leaks as the application runs, giving line numbers and call stack traces of the offending source. The Sentinel command is easily inserted into a makefile, and only a re-link is necessary to insert Sentinel into an application binary. Sentinel works with both C and C++ compilers.

Strengths:

- Easy to add or remove from an application
- Does not create any support files

Aids the developer by:

- Detecting dynamic memory errors
- Detecting stack corruption
- Detecting memory leaks
- Managing software watchpoints
- Providing debug tools with runtime features
- Supports many C/C++ compilers
- Sentinel is very inexpensive, its closest competitor costs ten times as much.

Weaknesses:

- It is unclear how well Sentinel works with C++ compilers. The current version seems to have difficulties with C libraries called from within C++ programs.
- Sentinel has a history of reporting memory error which are obviously not valid.

Overview:

Sentinel is an inexpensive, easy to use memory leak detector. Sentinel seems to have the functionality of leak detectors costing many times more.

Version: 1.3.5

Vendor Contact information:

Virtual Technologies, Inc.
46030 Manekin Plaza, Suite 160
Dulles, Virginia 20166
Phone: (703) 430-9247

Installation path:

/local/tools/sentinel

Invocation:

In Makefile, place -lsent on link line.

Licensing:

Unlimited use of site license.

Vendor documentation:

Items of interest:

Synopsys (Synopsys)

Synopsis:

Synthesis and simulation software for hardware design.

Category:

Hardware design tool.

Description:

VHDL System Simulator

Integrated simulation, logic synthesis, and test synthesis help create the design, validate the architecture, and analyze its behavior at a high level of abstraction. VHDL System Simulator enables the designer to capture concepts, verify high level specifications, and detect design inconsistencies before committing designs to gate level implementation.

Synthesis:

Synopsys synthesis translates and optimizes an HDL specification into a gate-level design according to the user-specified constraints. The synthesis product line consists of the following:

- HDL Compiler family - Synthesizes gate-level realizations from VHDL or Verilog HDL descriptions,
- Design Compiler family - synthesizes ASIC or FPGA layouts from gate-level descriptions,
- Design Analyzer - synthesizable, parameterizable designs (components) for use with Synopsys tools.

Library Compiler

Library Compiler helps users easily create and support ASIC and custom technologies by compiling library data into a format usable by Synopsys synthesis, simulation, and test tools.

Test Compiler Family

The Test Compiler family is a constraint-driven test synthesis tool that automates design-for-test and provides automatic test pattern generation (ATPG). The Test Compiler family integrates design for test, timing analysis, and ATPG to automatically explore trade-offs in performance, area, and testability.

DesignWare

DesignWare provides system developers a structured methodology for building an inventory of design knowledge suitable for integration in future product development cycles.

Strengths:

VHDL System Simulator

- Integrates closely with Synopsys synthesis tools.
- Supports designing at a high level of abstraction.
- Full design portability by supporting 100% of the VHDL language
- Supports high level design exploration and trade-off analysis.
- Complete set of integrated debugging tools
- Allows simulation of firmware on the modeled system for checking hardware/software interaction.
- Incorporated in Mentor Graphics Falcon Framework.
- Simulation Graphical Environment (SGE), a schematic capture/block diagram entry system, including ports and interconnect information. SGE then generates VHDL templates with entity descriptions for each block, as well as top level structure VHDL description.

HDL Compiler family:

- Provides architectural optimization
- Manages design complexity by allowing design specification at a higher level of abstraction
- Enhances productivity by automating gate-level implementation from high-level system descriptions
- Allows technology-independent design and design reuse

Design Compiler family:

- Performs constraint-driven logic optimization
- Reduces development cycle time
- Optimization of finite state machines for area, speed, state assignment, and slate minimization

Library Compiler

- Minimizes library maintenance by using a single library source for simulation, synthesis, and test. This feature also maintain data integrity and accuracy during translation for all Synopsys applications
- Speeds library development
- Enables quick, accurate updates and revision control

- Applicable for CMOS, FPGA, BiCMOS, GaAs and ECL technology libraries.

Test Compiler Family

- Shortens ASIC design cycle
- Minimizes cost of test
- Produces designs from either IEEE-1076 VHDL, Verilog, or existing design netlists.
- Eliminates the need for manual test structure insertion

DesignWare

- Facilitates design re-use
- Enables use of proprietary design data within high-level design while protecting intellectual property
- Reduces design time and risk
- Increases breadth of architectural evaluation

Weaknesses:

Overview:

Synopsys is a powerful tool for hardware design that supports design hierarchy. Various output options such as VHDL, Verilog, and MIF make Synopsys simple to integrate with other development tools. Synopsys is supported by a broad base of libraries which enables its simulation capabilities. VHDL Simulator provides a complete simulation environment for specifying and verifying electronic systems at the behavioral, RTL, and gate level.

Version: 3.0

Vendor Contact Information:

Synopsys, Inc.
700 East Middlefield Road
Mountain View, CA 94043-4033
Voice: (415) 962-5000
Fax: (415) 965-8637

Installation path:

Invocation:

Licensing:

Vendor documentation:

Items of interest:

TDS Wavemaker (TSSI)

Synopsis:

Wavemaker is an interactive graphics environment capable of creating, analyzing, modifying, and displaying both stimulus and response data.

Category:

Hardware Design and Test Support Tools

Description:

Wavemaker is a software package that allows design and test engineers to easily and accurately create and analyze waveform data for simulation and test program development. It provides four editors which are used to create, analyze, modify, and display stimulus and response data. Wavemaker separately generates and displays timing and pattern data for powerful and flexible manipulation of structured data.

Strengths:

- Wavemaker functions independently of both simulator and tester environments and can be used with a wide variety of popular CAE simulation and ATE test systems.
- Provides powerful, accurate analysis capabilities and automates many tedious, time consuming, and error prone tasks.
- Wavemaker can: compare simulations from different simulators
- compare different simulator models
- compare pre and post route simulations
- perform simulation rules checking
- perform tester rules checking
- edit timing pattern and waveform data

Weaknesses:

- The software package is not very user friendly.
- Both time and instruction are needed to get to the point of being able to use the tool effectively.

- There are some files that the user may have to create and others that the user may have to edit in order to be able to run some of the tools.

Overview:

Wavemaker can be a very important and useful tool but is hindered by the lack of ease of use. A lot of time may be required to learn to use the tool. Instruction may also be required in order to be able to use the tool effectively. Many of the steps needed to accomplish a task are not intuitive.

Version: 4.2.0.32

Vendor Contact Information:

TSSI
8205 SW Creekside Place
Beaverton, Oregon 97005
Phone: (503) 643 - 9281
Fax: (503) 646 - 4954

Installation Path:

/local/tools/tssi

Invocation:

A script called 'wavemaker' is used to invoke the software.

Licensing:

Currently there are two floating licenses.

Vendor Documentation:

The documentation consists of several thick volumes detailing the operation of the many different tools and utilities contained within the TSSI TDS software system. Wavemaker is just one of these many tools. The documentation is written using many acronyms and is somewhat difficult to use as simply a reference without previously having read the complete entity.

Items of Interest:

TDS Software System (TSSI)

Synopsis:

TSSI offers a software system with tools that automate the generation, analysis, management, and application of behavioral data for electronic circuits and systems.

Category:

Hardware Design and Test Support Tools.

Description:

TSSI accepts files from more than thirty simulators and systems, and provides tools for viewing, comparing, analyzing, and editing the resulting waveforms. When the waveform/ behavioral data is ready, TDS can produce test programs for more than eighty models of verification and component testers, and in-circuit and functional board testers.

Strengths:

- Captures simulation results from virtually any simulation environment
- Allows interactive viewing of simulation waveforms and measures capability in a simulator-independent format
- Can create, compare, and analyze composite waveforms for conformances to design specifications
- Can check simulation waveforms for conformance to design specifications
- Identifies abnormal bus behavior
- Automatically generate functional test programs
- Adds test specs to simulation waveforms
- Allows debug of test programs off line

Weaknesses:

- The software package is not very user friendly.
- Both time and instruction are needed to get to the point of being able to use the tool effectively.

- There are some files that the user may have to create and others that the user may have to edit in order to be able to run some of the tools.

Overview

The TSSI TDS software system is very powerful and flexible tool but it is hindered by the lack of ease of use. A lot of time may be required to learn to use the tool. Instruction may also be required in order to be able to use the tool effectively. Many of the steps needed to accomplish a task are not intuitive.

Version: 4.2.0.32

Vendor Contact Information:

TSSI
8205 SW Creekside Place
Beaverton, Oregon 97005
phone: (503) 643 - 9281
fax: (503) 646 - 4954

Installation Path:

/local/tools/tssi

Invocation:

A script called 'wavemaker' is used to invoke the tool.

Licensing:

Currently there are two floating licenses.

Vendor Documentation:

The documentation consists of several thick volumes detailing the operation of the many different tools and utilities contained within the TSSI TDS software system. The documentation is written using many acronyms and is somewhat difficult to use as simply a reference without previously having read the complete entity.

Items of Interest:

VHDL Simulator (Vantage)

Synopsis:

VHDL Compiler and Simulator

Category:

Hardware Simulation

Description:

Vantage Spread Sheet is a full IEEE-1076 compatible VHDL (VHSIC Hardware Description Language) Simulation environment. User-written VHDL files are compiled using the VHDL Compiler and simulated using SimView. Simulation results can be stored in a database or viewed in waveform or tabular format.

Strengths:

- Full IEEE-1076 compatibility IEEE-1076 is the IEEE standard which describes VHDL. Many other vendors only support a subset of this standard. Vantage, however, supports the entire language including Text IO functions (ie. ability to read and write from external text files) and "Advanced Topics" such as generics, aliases, and generate statements.

- Full IEEE-1064 compatibility IEEE-1064 describes the standard VHDL signal type, "STD_LOGIC", which is a nine-state signal type (ie. design nets can be in one of nine states including '0', '1', or 'X'). The signal type the designer chooses to describe his/her design has profound impact on the versatility of the design, availability of functions, and speed. Many other vendors do not support IEEE-1064 or provide a poor implementation. Vantage fully supports the standard and even has optimized its compiler to speed up simulations using "STD_LOGIC".

- Simulation Speed With an average knowledge of speed-sensitive VHDL design it is possible to write code to describe entire systems and run full system simulations that run on the order of hours rather than days. The Vantage simulator has been optimized to run efficiently on HP and Sun platforms.

- Debug1076 Vantage has one of the best run-time debuggers around. Source code is called up on the screen interactively. Break points can be set directly on any desired line of code. The value and description of any signal can be determined merely by clicking on that signal in the code, and single step capability can be used to find run time errors.

- Import/Export Vantage has established interfaces with the following formats for loading (importing) or dumping (exporting) entire designs : VHDL EDIF Cadence Viewlogic Mentor 7.x and 8.x

- LMC Interface available Simulations using Vantage Spreadsheet can utilize LMC Smart Models for standard parts. Also actual hardware can be used in Vantage simulation with the LMC Hardware Modeler interface.

- Zycad XP (hardware accelerator) interface available This interface is available, but with the speed of the standard VHDL simulator it is only in extreme cases that hardware acceleration would be necessary (See weaknesses also).

- Concurrent Compiler (Distributed Compilation) For large designs which take long periods of time to compile, compilation can be distributed over multiple workstations on a network.

- C interface / STYX Compiler For additional speed savings on time critical functions the C interface can be used to compile C coded elements into the VHDL environment.

Weaknesses:

- Mentor Import/Export Capability for Version 8.x is unreliable The ability to import schematics from Mentor 8.x is not reliable at this time. Many design constructs available in Mentor schematics are not accepted by Vantage. Also export capability to Mentor 8.x requires design capture using the Vantage schematic design entry tool called SVIEW. SVIEW is an extremely bare-bones schematic entry tool. It is far easier to do all netlist interfacing to Vantage using VHDL (ie. generate VHDL structural code from the schematic entry tool and read directly into Vantage).

- Zycad XP Interface is unreliable. Zycad XP interface requires an EDIF input. Zycad and Vantage do not seem to agree on the "flavor" of edif used resulting in an incompatible interface in most cases.

Overview:

Vantage Spreadsheet is probably the most powerful and complete VHDL compiler and simulator available on the market. Vantage has done a good job of creating a fast VHDL simulator which makes top-level (behavioral) system simulation possible at the beginning of the design process and thus reduces design cycle times.

Version:

4.171 running on HP730's and better. Also available on Sun platform.

Vendor Contact Information:

Jeff Goldman Vantage Analysis Systems, Inc.

Lakeshore Towers
18101 Von Karman, Suite 350
Irvine, CA 92715
Tel. (818) 398-3857
Fax. (714) 251-6590

Installation Path:

/usr/vss_v4.171

Invocation:

Must set up environment variables as follows:
setenv VANTAGE_VSS "/usr/vss_v4.171" setenv SYSTYPE "bsd4.3" set path=($path $VANTAGE_VSS/pgm/dir $VANTAGE_VSS/pgm/lmgr)
Then to invoke type:
VSS

Licensing:

Floating licenses available

Vendor Documentation:

Vantage Spreadsheet User Guide Volumes One and Two, April 1992
Training Material: Vantage Spreadsheet Training VHDL Basics Training

Items of Interest:

- Motif-Based user interface pegged for future releases.
- New SpeedWave simulation engine which is up to 2 times faster than the present simulator is due out in '94.
- Company President David Coelho was on the committee which created the IEEE-1076 standard describing VHDL and has also written one of the definitive texts on VHDL named "The VHDL Handbook", Kluwer Academic Publishers, 1989.

VxWorks (Wind River)

Synopsis:

VxWorks real-time operating system and development environment

Category:

Operating Systems

Description:

VxWorks combines a high-performance real-time operating system with sophisticated networking facilities and a complete set of cross-development tools. The microkernel, known as wind, provides VxWorks with high-speed multitasking capability and efficient task management, synchronization, and timing facilities. Operating system primitives add dynamic memory management, quick interupt handling capability, optimized floating-point support, and a broad range of intertask communications facilities. Additional features include both pre-emptive and round-robin scheduling and a real-time I/O and file system. VxWorks supports a wide range of standards, including POSIX, ANSI C, and UNIX™ networking.

VxWorks development target tools include an interactive C-interpreter shell, run-time linking, symbolic and source language debugging and performance monitoring, and an extensive set of utility libraries. Host-based tools include cross-compilers and a remote source level debugger.

Strengths:

The VxWorks kernel, wind, provides fundamental operating system primitives to satisfy real-time system requirements. Wind provides for efficient task management through the following:

- multi-tasking, unlimited number of tasks (limited only by hardware memory parameters)
- pre-emptive and round robin scheduling
- fast, deterministic context switching
- 256 priority levels
-

Wind also provides for fast, flexible intertask communications:

- semaphores
- binary, for task synchronization and resource guarding
- counting, to manage multiple resources
- priority inheritance, eliminates priority inversion
- message queues
- POSIX pipes

- sockets
- POSIX signals
- shared memory

In addition, wind provides fast, efficient interrupt handling, optimized floating-point support, dynamic memory management, and system clock and timing facilities.
VxWorks supports network connections over a variety of media including Ethernet, backplane, serial lines through SLIP (Serial Line Internet Protocol), and hooks for custom network interfaces. Complete UNIX™- standard networking includes:

- BSD 4.3 Tahoe UNIX™ TCP/IP
- Sockets: raw (IP), datagram (UDP/IP), and stream (TCP/IP)
- Remote login (rlogin, telnet)
- Remote Procedure Calls (RPC)
- Network File Systems (NFS)
- File Transfer Protocol (ftp/tftp client and server)
- Remote command execution (rsh)
- Multiprocessing support
- Network login security
- Bootp booting protocol

The VxWorks development environment features:

- Full ANSI compliance
- Interactive, C-interpreter target shell
- Symbolic debugging and disassembly
- Performance monitoring
- Exception and signal handling
- Extensive kernel, task, and system information utilities
- Linker loader
- System symbol table
- GNU C compiler support
- Libraries of over 600 utility routines
- Remote source language debugger (VxGDB)

The I/O and local file system features the following:

- POSIX I/O system and directory handling
- Fast, flexible I/O system
- SCSI support
- MS-DOS and RT-11 file system compatibility
- Raw disk file system

VxWorks features flexible booting from ROM, local disk, or over the network. Its highly scalable design allows the developer to configure the system for a wide range of applications.

A number of accessory products are also available including:

- windX, X Windows client package with Motif to support graphical applications

- VxVMI, virtual memory management interface featuring text and kernel data protection and a powerful programmer interface
- VxMP, tightly coupled multi-processing package providing system- wide semaphores, message queues, and memory, transparent, high performance design, and scale applications across up to 20 CPU's
- VxSim, provides full VxWorks simulation on a SPARCstation
- BSP Porting Kit, allows custom hardware porting with complete documentation, template software, and a powerful validation suite
- WindC++ Gateway, provides access to ObjectCenter, a complete, state of the art environment for writing, understanding and maintaining C++ programs
- GNU Toolkit, an efficient platform for application development hosted on UNIX™ workstations
- Stethoscope, for real-time graphical monitoring, performance analysis, and data collection
- RTILib, a collection of utilities and tools to assist in debug
- MicroWorks, a compact high-performance real-time operating system and powerful development environment offering VxWorks features for real-time embedded applications where size and performance are critical
- VADSWorks, combines the Verdix Ada Development System with VxWorks, for applications where ADA is a requirement

Weaknesses:

- Wind River Systems tools are extensive but single source.
- VxWorks must be hosted on SUN, HP, RS-6000, DEC, SGI, or MIPS UNIX™ workstations. IBM PC Windows 3.1 support will be available in 1Q94.
- Individual target license required for each processor.
- Proprietary application binary interface (ABI).
- Security is still an issue.

Overview:

VxWorks offers a rich and fully integrated environment tuned for real-time development designed to speed the development of highly reliable applications. VxWorks supports the widest range of host and target architectures, and offers off-the-shelf turnkey integration with an extensive set of commercial and evaluation boards. Its open design is highly portable and complete across all supported processors allowing application migration with minimal effort.

Version:

VxWorks 5.0.1 (installed on SPCOT network) VxWorks 5.0.2b (installed on SPCOT network) VxWorks 5.1 (described above) VxWorks 5.1.1 (to be released)

Vendor Contact Information:

Paul Kusiak
Wind River Systems Northeast Sales Office

77 North Washington St
Boston, MA 02114
617-367-6567

Wind River Systems Corporate Headquarters
1010 Atlantic Avenue
Alameda, CA 94501
800-545-WIND
510-748-4100
Fax: 510-814-2010

Technical Support:
1-800-USA-4WRS
1-510-814-2164 (fax)
email:support@wrs.com

Installation Path:

/local/tools/vx5.0.1 /local/tools/vx5.0.2b

Invocation:

Invoke Makefile by typing make. Makefile contains compiler directives, directory paths, build options. Can be rather complicated for an entire VxWorks build or simple for individual application builds.

Licensing:

Object Developers License $27500 (single site, 10 network users, single target processor only, additional target licenses extra) 1 year maintenance $4000

Vendor Documentation:

VxWorks Users Manual VxWorks Reference Manual VxWorks Programmer's Guide individual accessory manuals online documentation applications notes

Items of Interest:

VxWorks is currently in use on numerous RISC R3000 and Motorola 68xxx platforms throughout Lockheed Sanders.

X-Designer (VI Corp)

Synopsis:

X Windows/Motif graphical user interface builder.

Category:

Prototype User Interface

Description:

X-Designer is a Motif-based graphical user interface (GUI) builder for the X Windowing system. It allows the user to interactively prototype a GUI from a menu-driven interface without writing any code. The user can design and test a complete GUI without a single compilation. X-Designer then generates C/C++ code suitable for compilation and integration with other application code.

Strengths:

- All interface changes are instantly presented to the user, so development time is dramatically decreased. Gone are the days where a single widget change would force a recompile.
- Code generation can be structured in a way that allows seamless integration of regenerated GUI code with application code at any point during application development.
- The interactive nature of X-Designer helps to foster a deeper understanding of the workings of the Motif widget set, allowing the user to produce more complex user interfaces.

Weaknesses:

- X-Designer is a fairly recent product, and has been known to crash from time to time.
- A few minor inconsistencies in widget behavior between those displayed in XDesigner and the actual compiled application have been noticed.
- X-Designer lacks some useful features, such as the ability to modify resources on a set of widgets.

Overview:

X-Designer is an excellent GUI builder which can significantly reduce application

development time.

Version: 3.0b

Vendor Contact information:

VI Corporation
47 Pleasant Street
Northampton, MA 01060
Phone: (413) 586-4144
Fax: (413) 586-3805
Email: vi@vicorp.com

Installation path:

/local/tools/XDesigner

Invocation:

/local/bin/XDesigner

Licensing:

Floating license, currently limited to 1 user.

Vendor documentation:

Items of interest:

X-Designer is developed by:

Imperial Software Technology
95 London Street
Reading
Berkshire RG1-4QA
United Kingdom
Phone: +44 734 587055
Fax: +44 734 589005

Xact/Xilinx (Xilinx)

Synopsis:

XACT is a Field Programmable Gate Array (FPGA) and a Erasable Programmable Logic Device (EPLD) suite of design tools.

Category:

Hardware Design

Description:

The Base XACT package provides schematic capture and simulation interfaces, design implementation tools and download hardware for low-complexity Xilinx devices. These devices include the complete XC7200/XC7300 EPLD families, the complete XC2000 FPGA family and the XC3000/XC3100 FPGA family of devices up to the XC3130. A special version is available for Viewlogic on the PC that includes the Viewdraw schematic editor and the Viewsim simulator. Xilinx can provide the following packages to form a complete suite of design tools.

- ***Libraries and Interface*** - Contains symbol libraries for specified schematic editor, simulation models with timing information from specified simulator and a program to translate between the schematic editor or simulators file format and the XNF file format.
- **Core Implementation** - Provides the software necessary to process a netlist file into a bit-map file that can be downloaded into a Xilinx device. Includes tools fro logic reduction, design rule checking, mapping, automatic placement and routing, interactive placement and routing, bit-stream generation and bit-map file generation.
- **Logic Synthesis Interface** - Provides the tools to use a third-party high-level description language and synthesis for Xilinx design entry.
- **X-BLOX Architectural Synthesis** - Permits entering FPGA designs as block diagrams using a familiar schematic editor. Using built in expert knowledge, X-BLOX software automatically optimizes the design to take full advantage of the unique features of the Xilinx FPGA architecture.
- **Xilinx ABEL** - Supports text-based design entry and netlist translation using ABEL high-level description language. The ABEL language supports different design styles including Boolean equations, truth tables, and encoded or symbolic state machines.
- **Parallel Download Cable** - Supports downloading of FPGA bitstreams and PROM files from the parallel port of IBM PCs and compatibles.
- ***XChecker Cable*** - Supports downloading of FPGA bitstreams and PROM files and readback of configuration data and internal node values. This cable uses the serial port of IBM PCs and compatibles as well as workstations.
- ***XC3000 Demonstration Board*** - Provides demonstration or prototype capability for XC3000/XC3100 family devices in 68 pin PLCC packages.

- ***XC4000 Demonstration Board*** - Provides demonstration or prototype capability for XC4000 family devices in 84-pin PLCC packages.

Strengths:

- Multi-Platform support
- Tightly integrated with Viewlogic Viewdraw and Viewsim, Mentor Graphics Design Architect and Quicksim II, and also with OrCAD SDT and VST
- Both menu driven and Command line driven
- Vendor highly responsive to customer needs
- Can point to any node and see signal and timing

Weaknesses:

- Very slow on instantiation
- Needs two separate mouse drivers if running with Viewlogic

Overview:

XACT is a good package for the design and development of Xilinx based chips. It has adequate multi-platform support (IBM, HP, and SUN) and it is also tightly integrated with other high end FPGA/EPLD design environments.

Version: 4.12

Vendor Contact Information:

Regional:

Paul Laity - Regional Sales Manager
15615 Alton Parkway
Suite 280
Irvine, CA 92718
714-727-0780

Local (Phoenix):

Lee Miller
Quatra
4645 S Lakeshore Dr.
Suite #1
Tempe, AZ 85282
602-820-7050

Installation path:

Invocation:

Licensing:

Vendor documentation:

The descriptions for XACT were taken directly from the vendor data book.

Items of interest:

Motorola CIDM has done a complete evaluation of XACT version 4.12 and the strengths and weaknesses were taken directly from their evaluation.